A FRAMEWORK OF BELIEF PROPAGATION AND GAME THEORY FOR COGNITIVE RADIO SECURITY AND ROUTING

A Dissertation
Presented to
the Faculty of the Electrical and Computer Engineering Department
University of Houston

in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy
in Electrical Engineering

by
Zhou Yuan
August 2012

A FRAMEWORK OF BELIEF PROPAGATION AND GAME THEORY FOR COGNITIVE RADIO SECURITY AND ROUTING

Zhou Yuan

Approved:

Chair of the Committee
Dr. Zhu Han, Associate Professor
Electrical and Computer Engineering

Committee Members:

Dr. Haluk Ogmen, Professor
Electrical and Computer Engineering

Dr. Wei-Chuan Shih, Assistant Professor
Electrical and Computer Engineering

Dr. Rong Zheng, Associate Professor
Computer Science

Dr. Lijun Qian, Associate Professor
Electrical and Computer Engineering
Prairie View A&M University

Dr. Suresh K. Khator, Associate Dean,
Cullen College of Engineering

Dr. Badrinath Roysam, Professor and Chairman,
Electrical and Computer Engineering

Acknowledgements

First and foremost, I would like to express my deepest gratitude to my advisor Dr. Zhu Han for his excellent guidance and continual support during the course of my degree. Working with him was a wonderful experience and his wise knowledge, constructive advice and constant encouragement that he shared during my stay at the University of Houston has been invaluable. He contributed significantly to both my research and my professional development. I would also like to thank my dissertation committee, Dr. Haluk Ogmen, Dr. Wei-Chuan Shih, Dr. Lijun Qian, and Dr. Rong Zheng, for their encouragement, insightful comments, valuable discussions, and accessibility.

During my graduate studies at the University of Houston, I have had the pleasure of meeting many students, who have helped me directly or indirectly in completing my studies and have made my Ph.D. a rewarding experience. I owe my thanks to them. In particular, I would like to thank my labmates, Yi Huang, Nam Nguyen, Lanchao Liu, Najmeh Forouzandeh Mehr, Mohammad Esmalifalak and Jia Meng. I appreciate all the helpful discussions that I had with them over the years. I also thank my close friends, who have become an inseparable part of my life.

I am deeply indebted to my parents and my family who have been a constant source of support and love throughout this degree and my life. Thank you for everything.

A FRAMEWORK OF BELIEF PROPAGATION AND GAME THEORY FOR COGNITIVE RADIO SECURITY AND ROUTING

An Abstract
of a
Dissertation
Presented to
the Faculty of the Electrical and Computer Engineering Department
University of Houston

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy
in Electrical Engineering

by
Zhou Yuan
August 2012

Abstract

With the advent of new high data rate wireless applications, as well as growth of existing wireless services, demand for additional bandwidth is increasing rapidly. Existing spectrum allocation policies of the Federal Communications Commission (FCC) prohibits unlicensed access to licensed spectrum, constraining them instead to several heavily populated, interference-prone frequency bands, which causes spectrum scarcity. However, it has been shown by several spectrum measurement campaigns that the current licensed spectrum usage across time and frequency is inefficient. Therefore, a concept of unlicensed users temporarily "borrowing" spectrum from incumbent license holders to improve the spectrum utilization, called dynamic spectrum access (DSA), is proposed. Cognitive radio is a communication paradigm that employs software-defined radio technology in order to perform DSA and others versatile, powerful and portable wireless transceivers.

Up until now, most existing works have focused on spectrum sensing and spectrum access, but very few have focused on the higher layer, which is very important for cognitive radio networks. In this dissertation, we use the framework of distributed game theory and belief propagation to explore the routing techniques and the security issues in cognitive radio networks. Firstly, a belief-propagation based defense strategy for the primary user emulation (PUE) attack in cognitive radio networks is proposed, which avoids the deployment of additional sensor networks and expensive hardware in the networks used in the existing literatures. The proposed algorithm can provide low computational complexity and fast execution speed, and the framework is flexible to incorporate to defeat various kinds of attacks for future extension. In the next section, a brand new network-layer attack, named routing toward primary user (RPU) attack, is discovered in cognitive radio networks, in which malicious secondary users will try to route the data to those secondary users which are closer to the primary users in order to increase interference to the primary users. This new attack is very difficult to detect because the malicious nodes may claim that those nodes, to which they forward the packets, behave dishonestly and cause problems in the data transmission. Also a belief-propagation based defense algorithm is proposed in which each node keeps a table recording the feedbacks from the other nodes on the route, exchanges feedback information, computes beliefs and detects the malicious nodes based on the final belief values. Simulation results show that the proposed defense strategy against the RPU attack is effective and efficient in terms of significant reduction in the delay and interference caused by the RPU attack. Finally, we propose a distributed routing algorithm using the network formation game to minimize the aggregate interference from the secondary users to the primary users while keeping the delay along the route low. The proposed distributed routing algorithm can avoid the problems in the centralized routing solution, such as the high cost for building the centralized coordinate nodes, high information-gathering delay, and system breakdown caused by the possible failures in the centralized nodes, and is practically implementable. Simulation results show that the proposed scheme finds better routes in terms of interference to the primary users compared to the shortest path scheme, and the distributed solution shows near optimum compared to the centralized solution. The proposed technologies concerning the security issues and the routing algorithms in cognitive radio networks can provide a lot of benefits to society, and will assist the public safety, emergency services, and first responders communities in enabling better communications access to the network, which could potentially translate into additional human lives being saved.

Table of Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction and Background

1.1 Scarcity of the Radio Spectrum

In this day and age, the demand for wireless spectrum is increasing drastically due to the emergence of mobile users and wireless applications. The scarcity of the available spectrum has become a serious problem. A large part of the spectrum has been segmented and rented to licensed users by national spectrum regulators, such as the Federal Communications Commission (FCC) in the United States, based on traditional spectrum allocation policies. Wireless spectrum assigned to licensed users via these policies can only be used by those users who maintain exclusive rights across the specified range of frequencies within a geographical area. In other words, only licensed users can use this spectrum allocation, while other unlicensed users are not permitted to access this spectrum block and transmit their signal in this frequency range. Unlicensed devices only have access to heavily populated and highly interference-prone frequency bands, such as the industrial, scientific and medical (ISM) bands [1, 2].

However, a study by the Spectrum Policy Task Force (SPTF) of the FCC has shown that some frequency bands are heavily used by licensed systems in particular locations and at particular times, but that there are also many frequency bands which are only partly occupied or largely unoccupied [3]. For example, spectrum bands allocated to cellular networks in the United States reach the highest utilization during work hours, but remain largely unoccupied from midnight until early morning [4]. In Figure 1.1, we show a wireless spectrum measurement across the 924 MHz to 948 MHz frequency band collected by the Wireless Innovation Laboratory of the Worcester Polytechnic Institute in Worcester, MA on July 11, 2007 [2]. Notice that less than half of the spectrum is occupied in Figure 1.1, making the rest of the unoccupied spectrum inefficiently utilized.

The major reason that leads to the inefficient use of the wireless spectrum is the spectrum licensing scheme itself. In the traditional spectrum allocation policy, the radio spectrum cannot

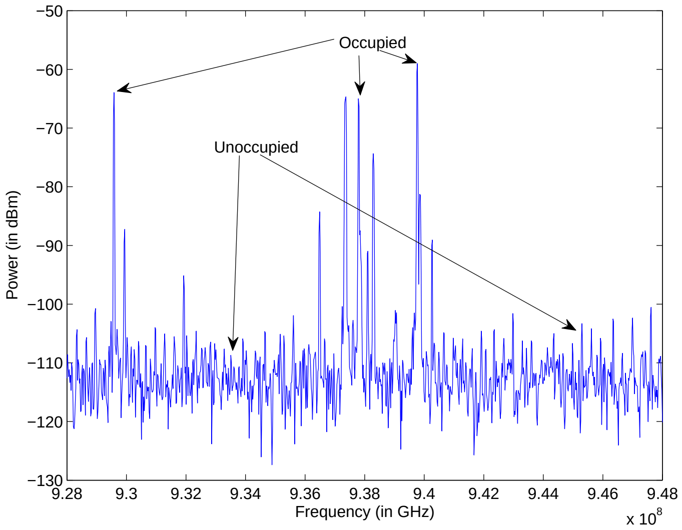

Figure 1.1 Spectrum occupancy measurements from 924 MHz to 948 MHz (7/11/2008, Worcester, MA, USA).

be utilized by unlicensed users and other applications even if licensed users are not occupying the spectrum. Due to this static and inflexible allocation, legacy wireless systems have to operate only on a dedicated spectrum band, and cannot adapt the transmission band according to the changing environment. This greatly decreases the efficiency of the wireless spectrum usage, and a new policy needs to be developed to improve this situation.

1.2 Dynamic Spectrum Access

To make better use of the wireless spectrum, a new concept called dynamic spectrum access (DSA) is proposed, in which unlicensed users can temporarily “borrow” spectrum from incumbent license holders. The basic concept is that unlicensed users decide on whether or not a particular frequency band is currently being used by licensed users and transmit the signal in that unused licensed band, while ensuring that the system performance of the primary users[1] as well as the secondary users[2] is not impacted. DSA can be defined as a mechanism to adjust the spectrum resource usage in a near-real-time manner in response to the changing environment and objective,

[1] Note that the terms primary user and licensed user are used interchangeably.

[2] Note that the terms secondary user and unlicensed user are used interchangeably.

changes of radio state, and changes in environment and external constraints [5].

There are three major models of dynamic spectrum access, namely, commons-use, shared-use, and exclusive-use models [6]. In the commons-use model, the spectrum is open to all wireless users. This model is already in use in the ISM band [7]. In the shared-use model, licensed users are allocated the frequency bands which are opportunistically accessed by the unlicensed users when the bands are not occupied by the licensed users. Unlicensed users have to guarantee that their transmissions will not interfere with the licensed signals. In the exclusive-use model, a licensed user can grant access of a particular frequency band to an unlicensed user for a certain period of time [8]. This model is more flexible than the traditional spectrum licensing model, since the type of use and the licensee of the spectrum can be dynamically changed.

With dynamic spectrum access, an unlicensed user can exploit unused in-band segments without causing interference to the active licensed users. There are two approaches for dynamic spectrum access: spectrum underlay and spectrum overlay [9]. The spectrum underlay approach constrains the transmission power of unlicensed users so that they operate below the interference temperature limit of the licensed users. One possible approach is to transmit the signals in a very wide frequency band (e.g., UWB communications) so that a high data rate is achieved with extremely low transmission power. It is based on the worst-case assumption that the licensed users transmit all the time. Therefore, it does not exploit spectrum white space (i.e., spectrum holes). The spectrum overlay approach, (or opportunistic spectrum access), does not necessarily impose any severe restriction on the transmission power by secondary users. It allows unlicensed users to identify and exploit the spectrum holes defined in space, time, and frequency. This approach is compatible with the existing spectrum allocation policy, and therefore, the legacy systems can continue to operate without being affected by the cognitive radio users. However, the basic etiquette for unlicensed users needs to be defined by the regulatory bodies to ensure compatibility with legacy systems.

1.3 Cognitive Radio

Dynamic spectrum access can be achieved by using a cognitive radio (CR), which is an autonomous unit in a communications environment that can determine the appropriate transceiver parameters based on its interaction with the environment, to enable secondary utilization of the spectrum. Cognitive radio is implemented based on software-defined radio (SDR), which is a reconfigurable wireless communication system in which the transmission parameter can be changed dynamically by software instead of hardware, such as operating frequency band, modulation scheme, and communication protocol. Cognitive radio can provide intelligent spectrum sensing, spectrum management, and spectrum access for unlicensed users. Cognitive radio can also learn from the environment and adapt its internal states to statistical variations in the existing RF stimuli by adjusting the transmission parameter, such as frequency band, modulation scheme, and transmit power, in a real-time manner [1, 10–15]. A cognitive radio network enables us to establish communications among cognitive radio users. A cognitive radio wireless network can be seen as a multi-channel multi-access network, and the wireless users work as secondary users who can opportunistically utilize spectral holes as far as they can guarantee that their transmissions will not interfere with primary users' signals [16–20]. The communication parameters can be adjusted according to change in the environment, topology, operating conditions, or user requirements. The use of a cognitive radio network provides a number of advantages when compared to cognitive radios operating purely autonomously:

- Improved spectrum sensing: By using cognitive radio networks, it is possible to gain significant advantages in terms of spectrum sensing.
- Improved coverage: By setting up cognitive radio network, it is possible to relay data from one node to the next. In this way power levels can be reduced and performance maintained.

1.3.1 Cognitive Radio Architecture

In the cognitive radio, the RF sections in the physical layer need to be particularly flexible. Not only may they need to swap frequency bands, possibly moving between portions of the radio communications spectrum that are widely different in frequency, but they may also need to change between transmission modes that could occupy different bandwidths [21]. To achieve the required level of performance, we need to develop a very flexible front end, which can be implemented based on software-defined radio. Traditional front end technology cannot handle these requirements because they are generally band limited, both for the form of modulation used and the frequency band in which they operate. Even so called wide band receivers have limitations and generally operate by switching front ends as required. Accordingly, the required level of performance can only be achieved by converting to and from the signal as close to the antenna as possible. In this way no analog signal processing will be needed, all the processing being handled by the digital signal processing. The conversion to and from the digital format is handled by digital to analog converters (DACs) and analog to digital converters (ADCs). To achieve the performance required for a cognitive radio, not only must the DACs and ADCs have an enormous dynamic range, and be able to operate over a very wide range, extending up to many GHz, but in the case of the transmitter, they must be able to handle significant levels of power.

In addition, the adaptive protocols in the MAC, network, transport, and application layers should be aware of the variations in the cognitive radio environment. In particular, the adaptive protocols should consider traffic activity of primary users, transmission requirements of secondary users, and variations in channel quality, etc. To link all modules, a cognitive radio control is used to establish the interfaces among SDR transceiver, adaptive protocols, and wireless applications and services. This cognitive radio module uses intelligent algorithms to process measured signal from the physical layer, and receive information on transmission requirements from the applications to control the protocols parameters in the different layers [1].

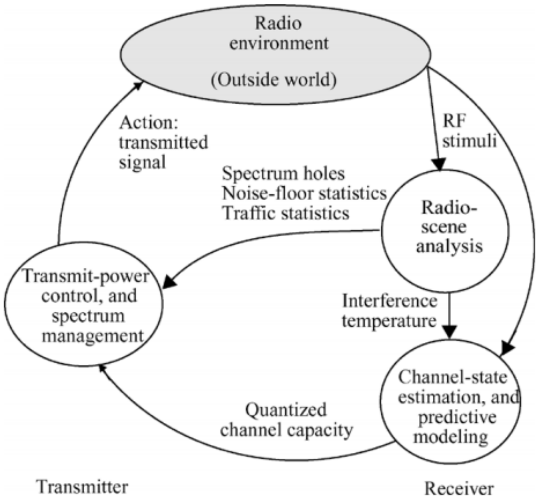

Figure 1.2 Characteristics and capabilities of cognitive radio system.

1.3.2 Capabilities of Cognitive Radio

There are three major technical features that characterize an intelligent cognitive radio, which is also shown in Figure 1.2:

- *The capability to obtain the knowledge of its radio operational and geographical environment.* Cognitive radio performs spectrum sensing mechanism to determine the status of the spectrum and the activity of licensed users by periodically sensing the target frequency band. In particular, a cognitive radio transceiver detects an unused spectrum or spectrum hole (i.e., band, location, and time) and also determines the method of accessing it (i.e., transmit power and access duration) without interfering the transmission of a licensed user.

  Spectrum sensing can be either centralized or distributed. In centralized spectrum sensing, a sensing controller (e.g., access point or base station) senses the target frequency band, and information obtained from sensing is shared to other nodes in the system. Centralized spectrum sensing can reduce the complexity of user terminals, since all the sensing functions are performed at the sensing controller. However, the centralized spectrum sensing suffers from location diversity. For example, the sensing controller may not be able to detect an unlicensed user at the edge of the cell. In distributed spectrum sharing, unlicensed users perform spectrum sensing independently, and the spectrum sensing results can either be used by individual cognitive radios (i.e., noncooperative sensing) or can be shared among other users (i.e., cooperative sensing). Although cooperative sensing incurs communication and processing overhead, the accuracy of spectrum sensing is higher than that of noncooperative sensing.

- *The capability to dynamically and autonomously adjust its operational parameters and protocols according to the knowledge, and to make decisions.* The information obtained from spectrum sensing is used to schedule and plan spectrum access by the unlicensed users. In this case, communication requirements of unlicensed users are also used to optimize the transmission parameters. Major components in spectrum management mechanisms are spectrum analysis and spectrum access optimization. In spectrum analysis, information from spectrum sensing is analyzed to gain knowledge about the spectrum holes (e.g., interference estimation, duration of availability, and probability of collision with licensed user due to sensing error). Then, a decision to access the spectrum (e.g., frequency, bandwidth, modulation mode, transmit power, location, and time duration) is made by optimizing the system performance given the desired objective (e.g., maximize throughput of unlicensed user) and constraints (e.g., maintain the interference caused to the licensed users below the target threshold).

  After a decision is made on spectrum access based on spectrum analysis, the spectrum holes are accessed by the unlicensed users. Spectrum access is performed based on a cognitive medium access control (MAC) protocol which intends to avoid collision with licensed users and also with other unlicensed users. The cognitive radio transmitter is also required to perform negotiation with the cognitive radio receiver to synchronize the transmission so that the transmitted data can be received successfully. A cognitive MAC protocol could be based on a fixed allocation MAC (e.g., FDMA, TDMA, CDMA) or a random access MAC (e.g., ALOHA, CSMA/CA).

- *The capability to learn from the results of its actions in order to further improve its perfor-*

mance. Machine learning algorithms from the field of artificial intelligence can be applied for learning in cognitive radio systems. A knowledge base of the spectrum access environment is built and maintained, which can be used to optimize and adapt the transmission parameters to achieve the desired objective under various constraints.

1.3.3 Spectrum Sensing

Spectrum sensing is one of the most key features for cognitive radio systems, which is to detect the presence of transmissions from the licensed users. There are three major types of sensing techniques, namely: multi-dimensional spectrum sensing, enabling spectrum sensing, and cooperative sensing [22].

Multi-Dimensional Spectrum Sensing

Conventional sensing methods usually relate to sensing the spectrum in the three dimensions: frequency, time, and space. However, there are other dimensions that need to be explored further for spectrum opportunity, such as the code dimension of the spectrum space. If the code dimension is interpreted as part of the spectrum space, we will know how to deal with signals that use spread spectrum, time or frequency hopping codes, and new opportunities for spectrum usage can be created. Another example is the angle dimension, which can create new opportunities for spectral estimation where not only the frequency spectrum but also the angle of arrivals (AoAs) need to be estimated. Therefore, it is of crucial importance to define such an n-dimensional space for spectrum sensing. However, this brings about new challenges for detection and estimation techniques, such as hardware requirements, security problems, hidden primary user problem, etc.

Enabling Spectrum Sensing

A number of different methods are proposed for identifying the presence of signal transmissions. In some approaches, characteristics of the identified transmission are detected for deciding the signal transmission as well as identifying the signal type.

- *Energy detector based sensing:* Energy detector based approach is the most common way of spectrum sensing because of its low computational and implementation complexities. In addition, it is more generic (as compared to methods given in this section) as receivers do not need any knowledge on the primary users' signal. The signal is detected by comparing the output of the energy detector with a threshold which depends on the noise floor. Some of the challenges with energy detector based sensing include selection of the threshold for detecting primary users, inability to differentiate interference from primary users and noise, and poor performance under low signal-to-noise ratio (SNR) values.

  Therefore, we can distinguish between the following two hypotheses: 1. H_0: $y(t) = s(t)$; 2. H_1: $y(t) = s(t) \times h + w(t)$, where $s(t)$ is the signal to be detected, $w(t)$ is additive white Gaussian noise (AWGN) sample, h is the channel gain and $y(t)$ is the received signal. Here, H_0 and H_1 are defined as the hypotheses of not having and having a signal from a licensed user in the target frequency band, respectively [22].

- *Waveform-based sensing:* Known patterns are usually utilized in wireless systems to assist synchronization or for other purposes. Such patterns include preambles, midambles, regularly transmitted pilot patterns, spreading sequences etc. A preamble is a known sequence transmitted before each burst and a midamble is transmitted in the middle of a burst or slot. In the presence of a known pattern, sensing can be performed by correlating the received signal with a known copy of itself. This method is only applicable to systems with known signal patterns, and it is termed as waveform-based sensing or coherent sensing.

- *Cyclostationarity-based sensing:* Cyclostationarity feature detection is a method for detecting primary user transmissions by exploiting the cyclostationarity features of the received signals. Cyclostationary features are caused by the periodicity in the signal or in its statistics like mean and autocorrelation or they can be intentionally induced to assist spectrum sensing. The cyclostationarity based detection algorithms can differentiate noise from primary users signal. Furthermore, cyclostationarity can be used for distinguishing among different types

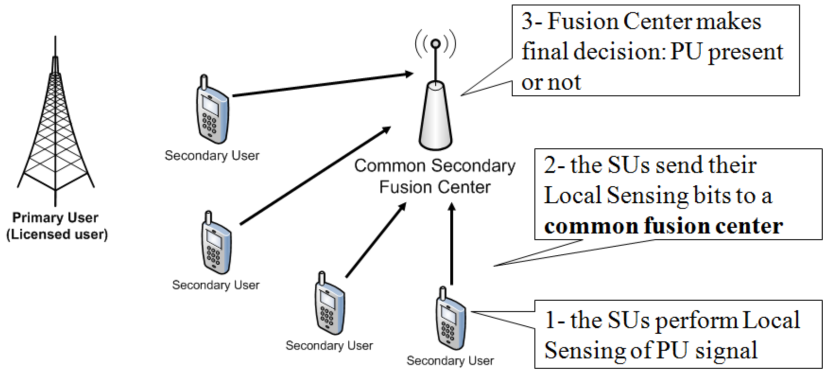
3- Fusion Center makes final decision: PU present or not
Secondary User
Common Secondary Fusion Center
2- the SUs send their Local Sensing bits to a **common fusion center**
Primary User (Licensed user)
Secondary User
Secondary User
Secondary User
1- the SUs perform Local Sensing of PU signal

Figure 1.3 An example for collaborative spectrum sensing.

of transmissions and primary users [22].

- *Matched-filtering:* The main advantage of matched filtering is the short time to achieve a certain probability of false alarm or probability of misdetection as compared to other methods. A matched filter will maximize the received SNR for the measured signal. Therefore, if the information of the signal from a licensed user is known (e.g., modulation and packet format), a matched filter is an optimal detector in stationary Gaussian noise. Matched filter detection is suitable when the transmission of a licensed user has pilot, preambles, synchronization word or spreading codes, which can be used to construct the template of the spectrum sensing [1].

Cooperative Sensing

Cooperative sensing decreases the probabilities of misdetection and false alarm considerably. In addition, cooperation can solve hidden primary user problems and it can decrease sensing time. The investigations on the interference to primary users caused by cognitive radio devices employing spectrum access mechanisms based on a simple listen-before-talk (LBT) scheme show that even simple local sensing can be used to explore the unused spectrum without causing interference to existing users. On the other hand, it is shown analytically and through numerical results that collaborative sensing provides significantly higher spectrum capacity gains than local sensing. The fact that cognitive radio acts without any knowledge about the location of the primary users in local sensing degrades the sensing performance.

- *Centralized sensing:* In centralized sensing, a central unit collects sensing information from cognitive devices, identifies the available spectrum, and broadcasts this information to other cognitive radios or directly controls the cognitive radio traffic.

- *Distributed sensing:* In the case of distributed sensing, cognitive nodes share information among each other but they make their own decisions as to which part of the spectrum they can use. Distributed sensing is more advantageous than centralized sensing in the sense that there is no need for a backbone infrastructure and it has reduced cost.

- *External sensing:* Another technique for obtaining spectrum information is external sensing. In external sensing, an external agent performs the sensing, and broadcasts the channel occupancy information to cognitive radios. External sensing algorithms solve some problems associated with the internal sensing where sensing is performed by the cognitive transceivers internally.

1.3.4 Spectrum Analysis and Spectrum Decision

Spectrum analysis is required for the characterization of different spectrum bands in terms of operating frequency, bandwidth, interference, primary user activity, and channel capacity. For example, in the spectrum underlay approach, based on the interference temperature limit at the primary receiver and operating frequency, the permissible transmission power at the cognitive radio can be determined. Subsequently, the channel capacity can be estimated. Spectrum analysis models can be based on either current spectrum sensing results or spectrum usage history.

The spectrum analysis architecture can be either local or cooperative. A cooperative architecture, which can be either centralized or distributed, can improve the accuracy of the spectrum usage model. A cooperative architecture requires exchanging information among cognitive radios (hence causes additional overhead) and may suffer from the scalability problem.

Spectrum decision deals with whether to transmit or not taking into account the fact that spectrum sensing/detection could be erroneous, and in case of transmission how to exploit the spectrum

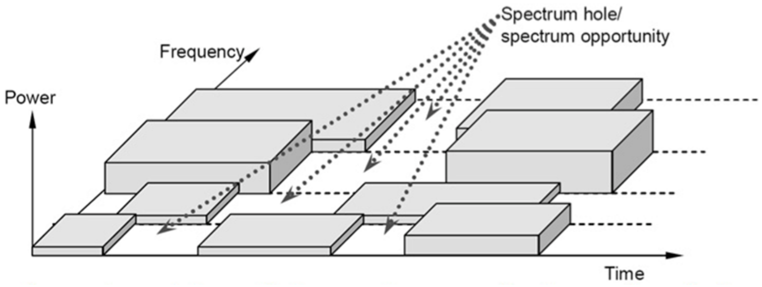

Figure 1.4 Spectrum access in time and frequency.

holes (i.e., what modulation and power level to use, how to share the spectrum holes among cognitive radios). This is primarily a medium access control problem for a cognitive radio and it also requires to consider the price offered/charged by the primary user/service provider. Also, spectrum access decisions may need to be communicated among the cognitive nodes and/or the intended receiver. Spectrum decisions can be made based on either a local or a global optimization criterion. In the case of local optimization, a spectrum access decision is made in a noncooperative (i.e., distributed) way and in the case of global optimization, a cooperative spectrum access decision is made either in a centralized or a distributed way.

In a noncooperative/local spectrum access strategy, each cognitive node is responsible for its own decision. If the missdetection probability is large, the access policy should be conservative. If the false alarm probability is large, the access policy should be aggressive. Therefore, access strategy can be jointly optimized with the sensing strategy. Game theory is a powerful tool to analyze the spectrum sharing problem in a noncooperative (i.e., competitive) spectrum access scenario. A noncooperative spectrum access strategy has minimal communication requirements (hence less overhead), but it may result in poor spectrum utilization.

In a cooperative centralized strategy, a centralized server maintains a database of spectrum availability and access information (based on the information received from a group of secondary users, say, through a dedicated control channel). Therefore, spectrum management is simpler and coordinated and it enables efficient spectrum sharing. Classical optimization theory can be used to solve the cooperative spectrum access problem in a centralized setting.

A cooperative distributed strategy relies on cooperative local actions throughout the network in order to achieve a performance close to the global optimal performance. However, this approach may suffer from the hidden node problem and large control overheads. Cooperative game theory techniques can be used to model and solve the spectrum access problem in such scenarios. In both the centralized and the distributed strategies, the primary user may or may not cooperate.

Once a decision is made to access the spectrum opportunities, several issues related to radio link control and resource management (i.e., how to access) need to be resolved. These include pulse shaping, transmission power control, selection of the number of spectrum bands to access and the set of appropriate bands (e.g., subcarriers in an OFDM system), adaptive modulation and coding (e.g., bit loading in the OFDM subcarriers) etc.

1.4 Routing in Cognitive Radio Networks

In traditional wireless networks, all network nodes will be provided with a certain fixed spectrum band for use. For instance, WLAN uses 2.4GHz and 5GHz bands, and GSM uses 900MHz and 1800MHz bands. In dynamic spectrum access networks, however, there may be no such pre-allocated spectrum that can be used by every node at any time, and the frequency spectrum that can be used for communication may vary from node to node. This new feature of cognitive radio networks imposes even greater challenges on wireless networking, especially on routing. If two neighboring nodes do not have a common channel, or they have common channels but do not tune to the same frequency, then multi-hop communication will not be feasible. Thus, new routing algorithms are needed in order to accommodate the spectrum dynamics and ensure satisfying network performance such as high network capacity and throughput, short latency, and lower packet loss [23].

1.4.1 Graph-Based Routing

Route design in classical wired/wireless networks has been tackled widely resorting to graph-theoretic tools. Graph-based routing can be easily extended into cognitive radio networks, given the full spectrum knowledge. We can design a two-phase approach for routing with a comprehensive framework to address channel assignment and routing jointly in semi-static multi-hop cognitive radio networks. The primary user dynamics are assumed to be low enough such that the channel assignment and the routing among secondary users can be statically designed. We focus on the case where cognitive devices are equipped with a single half-duplex cognitive radio transceiver, which can be tuned to M available spectrum bands or channels. The framework is based on the creation of a layered graph which features a number of layers equal to the number of available channels. Each secondary user device is represented in the layered graph with a node, A, and M additional subnodes, A_1, A_2, $\dots$, A_M, one for each available channel. The edges of the layered graph can be of three types: access, horizontal, and vertical. Access edges connect each node with all the corresponding subnodes. Horizontal edges between pairs of subnodes belonging to the same logical layer are added to the graph if the two corresponding secondary devices can be tuned to the corresponding channel. Vertical edges connect subnodes of different layers of a single secondary device, and represent the capability for a secondary device to switch from one channel to another to forward incoming traffic. The layered graph is a rather general framework which can be combined with different routing metrics [24]. Other graph tools, such as tree, can also be applied for routing problems in cognitive radio networks [25].

1.4.2 Power-Based Routing

In cognitive radio networks, sometimes the retrieval of information on spectrum occupancy is performed in a distributed way, and, similarly to classical ad hoc networks, distributed approaches are required to make local radio resource management decisions on partial information about the network state. In this subsection, we introduce power-based routing.

Minimum Power Routing

The main objective of minimum power routing is to discover minimum weight paths in cognitive wireless networks. The use of a Common Control Channel (CCC) plays a central role. A dedicated interface, referred to as Common Link Control Radio (CLCR) can be used for communication between cognitive radio terminals to sustain cognitive radio network related functions. The two main functions using CLCR interface are the neighbor discovery and path discovery and establishment. To discover a large neighborhood, CLCR uses a high transmission power to reach out to all the potential neighbors. A routing weight based on the required power to reach a specific destination is associated with different wireless systems. The minimum power routing protocol locally finds the path to minimize the routing weight between a source and a destination [26].

Bandwidth Footprint Minimization

Scheduling, power control, and routing problems can be addressed simultaneously in bandwidth footprint minimization (BFP) routing. The routing module is based on the notion of the bandwidth footprint product, in which the "footprint" refers the interference area of a node for a given transmission power. Since each node in the network uses a number of bands for transmission and each band has a certain footprint corresponding to its transmission power, the objective is to minimize network-wide BFP, which is the sum of BFPs for all nodes in the network. The proposed approach increases session rates with an iterative procedure. A conservative iterative procedure (CIP) and an aggressive iterative procedure (AIP) have been proposed to decide on the route selection, link scheduling, and the power allocation. CIP increases the rate of a session with the smallest scaling factor so as not to affect other sessions. On the other hand, AIP increases the rate of a session by allowing a limited decrease in other sessions' rate. Both CIP and AIP are composed of modules to determine the target decisions [27].

1.4.3 Delay-Based Routing

The quality of routing solutions can also be measured in terms of delays to establish and maintain multi-hop routes and to send traffic through the very same routes. Besides "classical" delay components for transmitting information in wireless networks, novel components related to spectrum mobility (channel switching, link switching) should be accounted for in multi-hop cognitive radio networks. In delay-aware routing we consider two different delay components:

- *Switching delay* that occurs when a node in a path switches from one frequency band to another.
- *Medium access delay* based on the MAC access schemes used in a given frequency band.

Solutions Accounting for Switching and Access Delay

Both the switching delay between frequency bands ($D_{switching}$) and backoff delay (medium access delay) within a given frequency band ($D_{backoff}$) should be considered when designing a routing algorithm for a multi-hop cognitive radio network. At relay node i, a metric representing the cumulative delay along a candidate route is computed as

$$D_{route,i} = DP_i + DN_i. \tag{1.1}$$

In the above equation, the first term DP_i takes into account the switching delay and backoff delay caused by the path and depends on the frequency bands assigned to all nodes along the path. As a consequence, we can have $DP_i = D_{switching,i} + D_{backoff,i}$. DN_i accounts for the switching and backoff delays caused by the existing flows at relay node i [28].

Solutions Accounting for Queuing Delay

$D_{switching}$ and $D_{backoff}$ can be integrated with a queuing delay arising at an intersecting relay node which serves n incoming flows [29]. The generalized cost function then becomes

$$C_{generalized} = D_{route,i} + D_{queueing}. \tag{1.2}$$

Analysis show that the queueing delay estimation is fairly accurate, and the end-to-end delay provided by the proposed routing protocol outperforms traditional routing solutions.

1.4.4 Throughput-Based Routing

In this subsection, we introduce the routing algorithms for cognitive radio networks with the main objective of throughput maximization.

Path Spectrum Availability Routing

The spectrum aware mesh routing (SAMER) proposal is a routing protocol that accounts for long term and short term spectral availability. SAMER seeks to utilize available spectrum blocks by routing data traffic over paths with higher spectrum availability, without ignoring instantaneous spectral conditions. The protocol first establishes candidate paths using periodically collected global states, and associating paths with path spectrum availability (PSA) metrics. Then, packets are delivered opportunistically along the path with the highest PSA value. SAMER seeks to utilize available spectrum blocks by routing data traffic over paths with higher spectrum availability. SAMER defines a metric for estimating PSA with the goal to capture:

- *Local spectrum availability*: Spectrum availability at a node i depending on the number of available spectrum blocks at i, their aggregated bandwidth and the contention from secondary users.

- *Spectrum blocks quality* depending on their bandwidth and loss rate.

The PSA is expressed as the throughput between a pair of nodes (i, j) across a spectrum block b,

$$Thr_{(i,j),b} = T_{f,b} \cdot B_{w,b} \cdot (1 - P_{loss,b}), \tag{1.3}$$

where $B_{w,b}$ is the bandwidth and $p_{loss,b}$ is the loss probability of the spectrum block b. $T_{f,b}$ is the minimum between the fractions of time during which the node $i(j)$ is free to transmit and/or receive packets through a spectrum block b. $Thr_{(i,j)}$ is the aggregate throughput between a pair of neighboring nodes which can be computed on the basis of the spectrum blocks available at a node i and then smoothed by multiplying by a value α to capture both the current view and the statistical information of spectrum availability [30, 31].

Spectrum Utility Based Routing

Opportunities to transmit are assigned based on the concept of spectrum utility and routes are explored based on the presence of spectrum opportunities with the objective of maximizing the spectrum utility. The spectrum utility for the generic link (i, j) can be defined as the maximum differential backlog between node i and node j,

$$U_{ij} = c_{ij}(Q_{i^{s*}} - Q_{j^{s*}}), \tag{1.4}$$

where c_{ij} is the achievable capacity for link (i, j), $Q_{i^{s*}}$ is the current backlog of packets at node i for the session (packet flow) s^* and s^* is the session with the highest differential backlog.

The generic node i performs the following actions:

- It periodically searches for the list of potential next-hops for session $s\{n_1, n_2, ..., n_N\}$.
- It calculates the capacity c_{ij}, where $j \in \{n_1, n_2, ..., n_N\}$ over the links towards all the potential neighbors; more specifically, given the current spectrum condition, each secondary user runs a distributed decision algorithm to decide which spectrum mini-bands should be used for the access and which power level to be used throughout the aforementioned spectrum bands.
- It chooses the actual next hop, j^*, that maximizes the spectrum utility, that is, $(s, j^*) = argmax_j(U^s_{ij})$.

1.4.5 Link Quality/Stability-Based Routing

The channel availability in multi-hop cognitive radio networks is significantly different from that in traditional wireless multi-channel multi-hop networks. Indeed, nodes in multi-hop cognitive radio networks potentially have partially overlapping or non-overlapping sets of available channels, and the available channel set at a secondary user is of time-varying nature and changes in a correlated or uncorrelated manner with respect to sets of other nodes. Consequently, network layer solutions should be able to cope with the necessity of re-routing in case specific portions of the currently active path are “impaired” by the presence of an activating primary user.

Solutions With Enhanced Path Recovery Functionalities

In the proposed framework with enhanced path recovery functionalities, the following need to be considered:

- Integrating spectrum discovery with route discovery to cope with spectrum heterogeneity;
- Having a coordination of the channel assignments of a per-flow basis, by minimizing inter-flow interference;
- Exploiting local spectrum heterogeneity to in order to have a spectrum diversity and reduce intra-flow interference.

To achieve these goals the solution starts the route set-up by broadcasting and AODV-style route discovery which accumulates information about each nodes available channels and their quality. At the end of the different paths towards the destination, each RREQ contains a list with the node IDs, the nodes’ spectrum availability and the links’ quality. Furthermore, to account for inter and intra flow interference, nodes intersecting different flows store the time schedules of these flows. The selected route is then reserved by using RREP messages. Channel usage is then scheduled at each node; a node can also locally change part of the channel assignment, in case of failures or node mobility, by keeping unchanged local throughput [32].

Solutions Targeting Route Stability

A route stability oriented routing analysis and a protocol are presented in [33], where a novel definition of route stability is introduced based on the concept of route maintenance cost. The maintenance cost represents the effort needed or penalty paid to maintaining end-to-end connectivity in dynamic multi-hop cognitive radio networks. The maintenance of a route may involve link switching and channel switching operations as a primary users become active. In the former case, one or more links along the route must be replaced by other ones not interfered with by primary users, whereas in the latter case, the same link can be maintained, but the transmission must be carried over to another spectrum portion. In either case, signalling is required to coordinate with other secondary users, which translates to a cost in terms of consumed power, and service interruption time while switching routes.

Owing to the heterogeneity of spectrum availability among nodes, the routing problem cannot be well solved without considering the spectrum allocation. The inter-dependence between route selection and spectrum management is studied in [34], where two design methodologies are compared. The first is a decoupled approach in which route selection and spectrum management are performed independently in different protocol layers. The second approach is a collaborative design, in which some tasks of spectrum management are integrated into route selection in the network layer. The network layer will select the packet route as well as decide a time schedule of a conflict-free channel usage. Experimental results show that a well-provisioned collaborative design outperforms the decoupled design.

In [35], the topology formation and routing in cognitive radio networks is studied. Nodes in the network first identify spectrum opportunities by detection, and then the detected spectrum opportunities are associated with the radio interfaces of each node. A layered graph model to help assign the spectrum opportunities to the the radio interfaces is proposed. Using the model, a routing path between nodes can be computed conveniently for each pair of nodes, which not only diversifies channel selection to prevent interference between adjacent hops along the path but also maximizes network connectivity.

1.4.6 Other Routing Approaches

There are some other routing algorithms for cognitive radio networks. In [36], the authors propose a distributed prediction-based routing algorithm that uses cognitive link availability prediction to predict the available duration of links. Based on the predictions, the algorithm constructs a topology which can improve throughput and delay. In [37] and [38], the authors apply swarm intelligence into the cognitive radio network routing problem, where [37] uses a reinforcement learning function to accelerate convergence, and [38] aims at minimization of delay. In [39], the authors focus on the routing algorithm based on the QoS requirements. A spectrum-aware cluster-based routing protocol where a spectrum-cluster is built in each spectrum band is proposed in [40].

1.5 Security Issues for Cognitive Radio Networks

New mechanism of spectrum access in cognitive radio systems brings many new dimensions of vulnerabilities. Due to the characteristics of cognitive radio systems, such as the requirement on the awareness of the surrounding environment and internal state, reasoning and learning from observations and previous experience to recognize environment variations, adaptation to the environment, and coordination with other users for better operation, cognitive radio networks face unique security challenges [23, 41]. People have discovered a few attacks in cognitive radio networks, and in the following subsections we summarize them in specific layers and their direct impact on the overall network performance.

1.5.1 Physical Layer Attack

In this subsection, we introduce the primary user emulation attack and reporting false sensing data attack for cognitive radio networks in the physical layer.

Primary User Emulation (PUE)

A fundamental characteristic of a cognitive radio is its ability for spectrum sensing, as it shall use the spectrum in an opportunistic manner. This means that the cognitive radio has to vacate a currently used spectrum band if an incumbent signal is detected. In this case, cognitive radios perform spectrum hand-off seeking for different spectrum holes for transmissions. Performing spectrum hand-off very often results in degradation of the cognitive radio performance since more time for sensing of the spectrum is required, and this decreases the available time for accessing the spectrum. This inherent operation of cognitive radios can be exploited by adversaries that mimic incumbent signals. Nodes launching primary user emulation (PUE) can be of two types:

- Greedy nodes that by transmitting fake incumbent signals force all other users to vacate a specific band (spectrum hole) in order to acquire its exclusive use.
- Malicious nodes (adversaries) that mimic incumbent signals in order to cause Denial of Service (DoS) attacks. Malicious nodes can cooperate and transmit fake incumbent signals in more than one band, thus causing extensive DoS attacks making a cognitive radio network hop from band to band, severely disrupting its operation. Furthermore, adversaries could also cause DoS attacks to PU networks by creating harmful interference.

Regardless of the type of the misbehaving node (greedy or malicious), the consequences to a cognitive radio network are the same: operation disruption and unfairness among the nodes. Normal secondary users may then believe that the primary user is present and avoid using the actually available spectrum bands (or channels) because of PUE attack. Recently, a more dangerous PUE is discovered, in which the attacker predicts which channel will be used by secondary users and attack these particular channels. Simulation shows that the PUE attack can increase the spectrum access failure probability from 10% (no attack) to 60% when there are 5 channels [42–44].

Reporting False Sensing Data

Reporting false sensing data (RFSD): Collaborative spectrum sensing is recognized as an efficient method to cope with the problem of unreliability in single-user spectrum sensing. However, false sensing data can cause false alarm or miss detection in the decision made by the fusion center (FC). Nodes sending false observations can be categorized as follows:

- Malicious users that send false observations in order to confuse other nodes or the FC. They aim to lead FC or the rest of the nodes to falsely conclude that there is an on-going incumbent transmission where there is not, or make them believe that there are no incumbent transmissions when there are. In the first case, the legitimate secondary users will evacuate the specific band, while in the second case they will cause harmful interference to the primary users.
- Greedy users that continuously report that a specific spectrum hole is occupied by incumbent signals. The goal of these users is to monopolize the specific band by forcing all other nodes to evacuate it.
- Unintentionally misbehaving users that report faulty observations for spectrum availability, not because they are malicious or greedy, but because parts of their software or hardware is malfunctioning. The reason for this can be a random fault or a virus.

Regardless of the type of the misbehaving users, the reliability of collaborative spectrum sensing can be severely degraded by faulty provided observations. This is called a reporting false sensing data attack. For instance, with a detection rate of 0.99%, a single malicious secondary user can increase the false alarm rate to 36% when it attacks only half of the time [45–48].

1.5.2 MAC Layer Attack

In this subsection, we introduce the attacks for cognitive radio networks in the MAC layer.

Common Control Channel (CCC) Attack

Common control channel (CCC) plays an important role in enabling cognitive radios to exchange control information. CCC is used for the exchange of several control information regarding for example collaborative sensing, channel negotiation, spectrum hand-off, etc. Protecting the CCC is very important, as this could be the first mechanism that a sophisticated adversary will try to compromise. In the event of a successful attack, network performance will be severely affected since CCC is the main mechanism for controlling network operations. The threats that a CCC faces can be categorized as follows:

- MAC spoofing, where attackers send spurious messages aiming to disrupt the operation of a cognitive radio network (e.g., channel negotiation). Multi-hop cognitive radio networks are more vulnerable to this type of attack as there is no central entity to control the authentication between the nodes and protect data integrity.

- Congestion attacks, where attackers flood the CCC in order to cause an extended denial of service (DoS) attack.

- Jamming attacks, where attackers launch DoS attacks by creating interference or sending superfluous packets, such that legitimate secondary users have less chance to find common available channels and therefore less chance to communicate with each other.

The degree of degradation of performance of a multi-hop cognitive radio network is heavily affected by the number of the attackers. In a multi-hop cognitive radio network, selfish nodes that are located along the path of normal-behaved nodes can drop their packets; thus monopolizing the medium. The results show that with a given topology, the throughput can drop by 20% when the percentage of selfish nodes becomes more than 25%.

Reporting False Selection Frame

When two secondary users want to establish a communication channel, the sender first sends a Free Channel List (FCL) frame to the receiver, then the receiver will respond with a SELection (SEL) frame to indicate the data channel they are going to use. Upon receiving the SEL frame, the sender will notify its neighbors of the channel selection via a channel reservation message (RES frame). In this process, a selfish secondary user may always claim that there is no available channel and refuse to forward the package for other nodes.

Recent studies show that reporting false selection frame can seriously degrade a cognitive radio network's performance [49]. To measure the network fairness, Jain's fairness index can be used, which is defined as

$$Fairnessindex = \frac{\left(\sum_f R_f\right)^2}{n\sum_f R_f^2}, \tag{1.5}$$

where R_f is the measured throughput of each data flow and n is the total number of flows in the cognitive radio network. Analysis results show that the network fairness can be greatly decreased because of the reporting false selecting frame attack.

False Evacuation

If the primary user turns on its transmission while secondary users are transmitting, secondary users need to evacuate the channel through an evacuation protocol. A malicious node can attack the evacuation protocol by sending false warning information to other secondary users, to force evacuation even if the primary user is off [50]. We assume that the adversary knows the parameters of the warning message (i.e., the pattern of the warning message, the CDMA spreading code to be used and the transmission power) for any evacuation group in the system. It can learn these parameters by eavesdropping during the initialization phase or analyzing the warning messages sent during the normal operation phase. Suppose that a given channel is idle and used by secondary users in an opportunistic manner. The adversary can deprive the secondary users from using this channel by sending a fraudulent warning message. The warning message will be relayed by other secondary users and the channel will be quickly evacuated although it is idle. By repeating this procedure, the adversary can easily trick the secondary users to often evacuate the channels and spend most of their time searching for available spectrum instead of engaging in communication with other nodes in the network. An adversary can also mount a limited attack on the primary network. Suppose a primary user returns and starts using a channel that is used by some secondary users as well. If the compromised nodes do not report the return of the primary user and do not relay the warning messages generated by other secondary users, then it is possible that some of the secondary users will not be aware that the primary user has returned, and continue to use the channel even though it is no longer idle.

1.5.3 Network Layer Attack

Many higher layer attacks of other traditional wireless networks could also apply to cognitive radio networks. In the network layer, the most prominent attacks are selective forwarding (i.e., blackhole/grayhole), wormhole, and sybil attacks. In a blackhole attack, which is a redirection attack, the attacker induces the source node to choose a route through the attacker and the attacker can then misuse, eavesdrop or drop messages as it sees fit [51]. A wormhole attack is another kind of redirection attack, in which two colluding attackers have a high speed link between them. This will make other nodes wrongly believe that the path between the colluding attackers is much shorter than other paths. The colluding attackers can then attract a large amount of data traffic, which causes congestion or facilitates data manipulation and traffic analysis [52]. Another network layer attack is called a sybil attack, in which a malicious node behaves as if it were a larger number of nodes, for example by impersonating other nodes or simply by claiming false identities, aiming to gain a disproportionately large influence in the network [53, 54]. The goal of the various attacks is mainly to reduce the network throughput [55–58]. However, all of these attacks listed above are discovered in wireless mesh/sensor/ad hoc networks, without considering much about the cognitive radio system model and existence of the primary users.

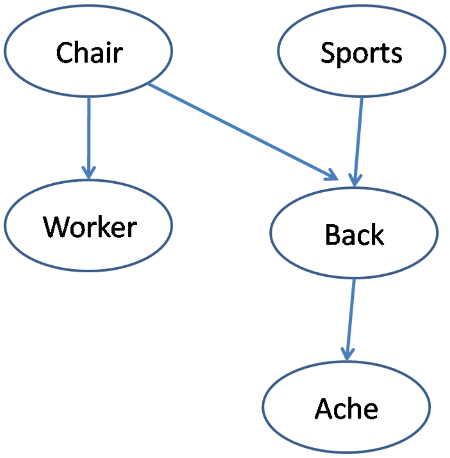

Figure 1.5 An example for Bayesian network.

1.5.4 Cross-Layer Attack

In the previous subsections we present threats focused only on single-layer measurements and techniques. However, adversaries can launch attacks targeting multiple layers. These are known as cross-layer attacks and can affect the whole cognitive cycle, as attacks at all layers become feasible.

One example is that RFSD and small-backoff-window (SBW) attacks can be launched together against a cognitive radio network. SBW is a very common attack in wireless networks where malicious users choose a very small value for a minimum contention window (CWmin) aiming to monopolize bandwidth. SBW attacks are feasible against cognitive radios with MAC layers using a carrier sense multiple access with collision avoidance (CSMA/CA) type of access. RFSD mainly attacks in the physical layer. The combination of these two attacks can contribute to significant performance degradation in cognitive radio systems.

1.6 Inference and Belief Propagation

1.6.1 Inference and Bayesian Networks

Inference problems arise in statistical physics, computer vision, error-correcting coding, and artificial intelligence. Bayesian networks are the most popular type of graphical model which can solve the problems including medical diagnosis, map learning, language understanding, and heuristic search [59]. Consider the following example that illustrates some of the characteristics of a Bayesian network shown in Figure 1.5. This example considers a person who might suffer from a back injury, an event represented by the variable Back (denoted by B). Such an injury can cause a backache, an event represented by the variable Ache (denoted by A). The back pain might result from a sports injury, represented by the variable Sports (denoted by S) or from new uncomfortable chairs installed in the person's office, represented by the variable Chair (denoted by C). In the latter case, it is reasonable to assume that a co-worker will suffer and report a similar backache syndrome, an event represented by the variable Worker (denoted by W). All variables are binary; thus, they are either true or false.

In this example the parents of the variable Back are the nodes Chair and Sports. The child of Back is Ache, and the parent of Worker is Chair. Following the Bayesian network independence assumption, several independent statements can be observed in this case. For example, the variables Chair and Sports are marginally independent, but when Back is given they are conditionally dependent. This relation is often called explaining away. When Chair is given, Worker and Back are conditionally independent. When Back is given, Ache is conditionally independent of its ancestors, Chair and Sport.

In our example, the overall or joint probability is the product of all the probabilities of the parent nodes and all the conditional probabilities,

$$p(\{x\}) = p(x_C)p(x_S)P(x_W|x_C)p(x_B|x_C, x_S)p(x_A|x_B). \tag{1.6}$$

More generally, a Bayesian network is a directed acyclic graph of N random variables x_i that defines a joint probability function

$$p(x_1, x_2, \cdots, x_N) = \prod_{i=1}^{N} p\left(x_i|Par(x_i)\right), \tag{1.7}$$

where $Par(x_i)$ denotes the states of the parents of node i. If node i does not have parents, we take $p\left(x_i|Par(x_i)\right) = p(x_i)$.

Our goal is to compute the marginal probabilities. For example, we might want to compute the probability that a patient has a backache. By inference, we simply mean the computation of these marginal probabilities. Mathematically, the marginal probabilities are defined in terms of sums over all the possible states of all the other nodes in the system. For example, the marginal probability of the last node $p(x_N)$ can be calculated as

$$p(x_N) = \sum_{x_1} \sum_{x_2} \cdots \sum_{x_{N-1}} p(x_1, x_2, x_3, \cdots, x_N). \tag{1.8}$$

We refer to the marginal probabilities that we compute approximately as “beliefs,” and denote the belief at node i by $b(x_i)$.

For small Bayesian networks, we can easily do marginalization sums directly, but unfortunately, the number of terms in sums will grow exponentially with the number of hidden nodes in the network. The virtue of the belief propagation algorithm is that we can use it to compute marginal probabilities, at least approximately, in a time that grows only linearly with the number of nodes in the system. For that reason, belief propagation can be used in practice as an “inference engine” acting on the statistical data encoded in a large Bayesian network.

1.6.2 Pairwise Markov Random Fields

People often take for granted the solution of apparently simple computer vision problems like the segmentation and recognition of objects or the detection and interpretation of motion. However, it is difficult for a us to teach a computer to solve such problems given just a series of two-dimensional arrays of pixel values. A key to progress in computer vision is to find theoretically solid models that are computationally tractable.

Pairwise Markov random field (MRF) provides an attractive theoretical model for such problems, in which we normally want to infer a representation from the data that we are given. For example, suppose that we want to infer the distance of the objects in a scene from the viewer. Imagine that we are given a 1000 by 100 gray-scale image, and we pose our problem as trying to infer distance value d_i corresponding to intensity values I_i, where i ranges over the million possible

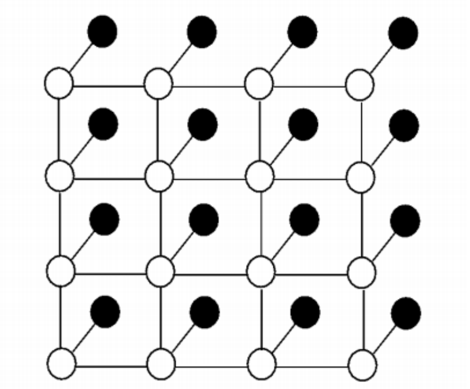

Figure 1.6 Pairwise Markov random field.

pixel positions. Or instead, we might be inferring some other quantity about the scene, such as high-resolution details that are missing in the image, or the optical flow in a series of images, etc.

In general, we assume that we observe some quantities about the image y_i, and that we want to infer some other quantities about the underlying scene x_i. If we have some information about some of the nodes, we will call such a node an "observable" node, in contrast to the other nodes which are "hidden" nodes. The indices i could represent single pixel positions, or they might represent the position of a small patch of pixels. We further assume that there is some statistical dependency between x_i and y_i at each position i, which we define as a local function $\phi_i(x_i, y_i)$. The function $\phi_i(x_i, y_i)$ is often called the evidence for x_i. In addition, for us to possibly be able to infer anything about the scene, there has to be some structure to the x_i. We encode the structure of the scene by saying that the nodes are arranged in a two-dimensional grid, and scene variable x_i should be compatible with nearby scene variables x_j. Then we have a compatibility function $\psi_{ij}(x_i, x_j)$ which connects nearby positions. We can calculate the overall joint probability of a scene x_i and an image y_i as

$$p(\{x\}, \{y\}) = \frac{1}{Z} \prod_{ij} \psi_{ij}(x_i, x_j) \prod_{i} \phi(x_i, y_i), \tag{1.9}$$

where Z is a normalization constant.

A graphical depiction of MRF is shown is Figure 1.6. The filled-in circles represent the observed node y_i, while the empty circles represent the hidden node x_i. The Markov random field is said to be pairwise because the compatibility function only depends on the pairs of i and j. In contrast to the Bayesian network, MRF is undirected. There is no notion, as is usually implicit in a Bayesian network, that the variable at one node x_i is a causal parent of its neighbor x_j, so we use undirected compatibility function $\psi_{ij}(x_i, x_j)$ instead of conditional probability function $p(x_i|x_j)$. Nevertheless, our agenda in doing inference will be very similar: we want to compute the belief $b(x_i)$, for any position i, so as to be able to infer something about the underlying unknown scene. Once again, a direct computation of marginal probabilities would take exponential time.

1.6.3 Belief Propagation

In this subsection, we develop belief propagation framework based on pairwise Markov random field. We consider the observed node y_i to be fixed, write $\phi_i(x_i)$ as a short-hand for $\phi_i(x_i, y_i)$ and focus on the joint probability distribution for the unknown variable x_i,

$$p(\{x\}) = \frac{1}{Z} \prod_{ij} \psi_{ij}(x_i, x_j) \prod_{i} \phi(x_i). \tag{1.10}$$

In the belief propagation algorithm, we introduce a variable defined as $m_{ij}(x_j)$, which can intuitively be understood as a message from a hidden node i to hidden node j about what state node j should be in. The message $m_{ij}(x_j)$ will be a vector of the same dimensionality as x_j, with each component being proportional to how likely node i thinks it is that node j will be in the corresponding state. In the belief propagation algorithm, the belief at a node i is proportional to the product of the local evidence at that node ($\phi_i(x_i)$), and all the messages coming into node i,

$$b_i(x_i) = k\phi_i(x_i) \prod_{j \in N(i)} m_{ji}(x_i), \tag{1.11}$$

where k is a normalization constant (the beliefs must sum to 1) and $N(i)$ denotes the nodes neighboring i. The messages are determined self-consistently by the message update rules,

$$m_{ij}(x_j) \leftarrow \sum_{x_i} \phi_i(x_i)\psi_{ij}(x_i, x_j) \prod_{k \in N(i)\backslash j} m_{ki}(x_i). \tag{1.12}$$

Note that on the right hand side, we take the product over all messages going into node i except for the one coming from node j [59].

Belief propagation is a way of organizing the global computation of marginal beliefs in terms of smaller local computations. The message flow is akin to the flow of a river, and each message summarizes all the computations that occurred further upstream along the branches that feed into that message. The belief propagation algorithm defined in Eq. (1.11) and Eq. (1.12) does not make reference to the topology of the graph that it is running on. Thus, there is nothing to stop us from implementing it on a graph that has loops. One starts with some initial set of messages (one usually begins with completely unbiased messages), and simply iterates the message-update rules until they converge, and then one can read off the approximate beliefs from the belief equations. In addition, in belief propagation, the computational complexity increases only linearly with the number of nodes in the framework, which provides significant efficiency.

1.7 Contributions and Organization of this Dissertation

Although it has been more than one decade since the concept of cognitive radio was raised for the first time, the research on the security issues and the routing techniques on cognitive radio networks is still limited. In this dissertation, we explore the security issues in cognitive radio networks. We develop a defense strategy using belief propagation against the primary user emulation attack in cognitive radio networks, which is effective and efficient. Furthermore, we discover a brand new network-layer attack, named routing towards primary user attack. We analyze the damage caused by this attack, and we propose the corresponding defense algorithm using belief propagation. The reason we propose belief propagation based defense strategy is that belief propagation is an effective and efficient mathematical tool to calculate the marginal probability based on many local observations. In addition, we propose an innovative routing algorithm for cognitive radio networks considering the aggregate interference from the secondary users to the primary users while keeping the delay low. Network formation game is applied for the distributed algorithm, which provides a suitable routing framework because the network structures play an important role and determine the outcome of the interactions in the network formation process.

In Chapter 2, we propose a defense strategy against the primary user emulation (PUE) attack in cognitive radio networks using belief propagation, which avoids the deployment of additional sensor networks and expensive hardware in the networks used in the existing literatures. In PUE attack, malicious nodes will mimic incumbent signals and claim that they are primary users. In this case, other honest secondary users will provide those malicious nodes higher priority to access the spectrum. Primary user emulation is one of the most serious attacks for CR networks, which can significantly increase the spectrum access failure probability. In our proposed defense approach, each secondary user uses belief propagation to calculate the local function and the compatibility function, compute the messages, exchange messages with the neighboring users, and calculate the beliefs until convergence. Then, the PUE attacker will be detected, and all the secondary users in the network will be notified in a broadcast way about the characteristics of the attacker's signal. Therefore, all SUs can avoid the PUE attacker's primary emulation signal in the future. Simulation results show that our proposed approach converges quickly, and is effective to detect the PUE attacker.

In Chapter 3, we propose a new and powerful network layer attack, routing-toward-primary-user (RPU) attack in cognitive radio networks. In this attack, malicious nodes intentionally route a large amount of packets toward the primary users, aiming to cause interference to the primary users and to increase delay in the data transmission among the secondary users. In the RPU attack, it is difficult to detect the malicious nodes since the malicious nodes may claim that those nodes, to which they forward the packets, behave dishonestly and cause problems in the data transmission. To defend against this attack without introducing high complexity, we develop a defense strategy using belief propagation. Firstly, an initial route is found from the source to the destination. Each node keeps a table recording the feedbacks from the other nodes on the route, exchanges feedback information and computes beliefs. Finally, the source node can detect the malicious nodes based on the final belief values. Simulation results show that the proposed defense strategy against the RPU attack is effective and efficient in terms of significant reduction in the delay and interference caused by the RPU attack.

In Chapter 4, a distributed routing algorithm is proposed using the network formation game to minimize the aggregate interference from the secondary users (SUs) to the primary users (PUs). In cognitive radio networks, although the interference from a single SU that is outside the PUs' footprints is small, the aggregate interference from a great number of SUs transmitting at the same time may be significant, and this will greatly influence the PUs' performance. The proposed distributed algorithm uses the network formation game with the utility function considering the aggregate interference from the SUs to the PUs. The proposed algorithm can avoid the problems in the centralized approach, such as the high cost for building the centralized coordinate nodes, high information-gathering delay, and system breakdown caused by the possible failures in the centralized nodes. From the simulation results, we can observe that the proposed algorithm can find the routes with lower interference to the PUs compared to the Dijkstra's shortest algorithm. Compared to an upper bound, the distributed solution shows near optimum.

Finally, in Chapter 5, we conclude our work and explore the possible extensions of our proposed framework. We propose some future work, such as the extension for a cross-layer attack and defense, incorporate medial axis and dynamic game framework for routing, and real-world implementation in Universal Soft-ware Radio Peripherals 2 hardware.

Chapter 2

Defeating Primary User Emulation Attacks Using Belief Propagation in Cognitive Radio Networks

In this chapter, we propose a defense strategy against the primary user emulation (PUE) attack in CR networks using belief propagation, which avoids the deployment of additional sensor networks and expensive hardware in the networks used in the existing literatures. We first briefly review the primary user emulation attack in cognitive radio networks. Then in Section 2.2, we introduce the system model and naive received signal based detection strategy. However, the PUE attacker can use an advanced strategy to defeat the naive detection. In 2.3, we present the detection strategy by simple interactions between neighboring users. In order to overcome practical challenges, in Section 2.4 the defense strategy against a PUE attack using belief propagation is developed, and the complete algorithm is presented. Each secondary user calculates the local function and the compatibility function, computes the messages, exchanges messages with the neighboring users, and calculates the beliefs until convergence. Then, the PUE attacker will be detected, and all the secondary users in the network will be notified in a broadcast way about the characteristics of the attacker's signal. Therefore, all SUs can avoid the PUE attacker's primary emulation signal in the future. Section 2.5 presents the simulation results, which show that our proposed approach converges quickly, and is effective to detect the PUE attacker. Finally, in Section 2.6, we conclude the chapter.

2.1 Primary User Emulation Attack

Cognitive radio (CR) is a promising technology for future wireless spectrum allocation to improve the usage of the licensed bands. However, CR wireless networks are susceptible to various attacks and cannot offer efficient security. Primary user emulation (PUE) is one of the most serious attacks for CR networks, which can significantly increase the spectrum access failure probability.

In a PUE attack, the malicious nodes emulate the feature of the primary user's signal characteristics and transmit in available secondary spectrum when PUs are inactive in CR networks. As a result, the naive secondary users may believe that the PUs are present and avoid using the actually available spectrum bands (or channels).

Some recent work has tried to analyze and defend the PUE attack. In [60], the authors proposed an analytical approach and obtained a lower bound on the probability of a successful PUE attack on a secondary user in a cognitive radio network by a set of co-operating malicious users. They showed that the probability of a successful PUE attack increases with the distance between the primary transmitter and secondary users. Two localization-based defense strategies against the PUE attack were proposed in [42] and [61]. In [42], RSS-based localization was used to determine the location of the attacker by deploying an additional sensor network. The authors employed a non-interactive localization scheme to locate the attacker. In [61], a joint position verification method using both time difference of arrival (TDOA) and frequency difference of arrival (FDOA) was proposed to enhance the positioning accuracy. In [62], the authors showed that both the attacker and the defender can apply estimation techniques and learning methods to obtain the key information of the environment, and thus better strategies can be developed. In [63], the authors analyzed the PUE attack problem within a Bayesian game framework, in which users are unsure of the legitimacy of the claimed type of other users. They showed that depending on radios' beliefs about the fraction of PUs in the system, a policy maker can control the occurrence of emulation attacks by adjusting the gains and costs associated with performing or checking for emulation attacks. In [64], the authors integrated cryptographic signatures and wireless link signatures (derived from physical radio channel characteristics) to enable primary user detection in the presence of attackers. In [65], the authors proposed a method that allows primary users to add a cryptographic link signature to its signal so the spectrum usage by primary users can be authenticated. We proposed a belief propagation based model to detect a PUE attack in [66]. However, the system model was preliminary and the compatibility function was considered only as a constant.

In this chapter, we propose a new received signal strength (RSS)-based defense strategy

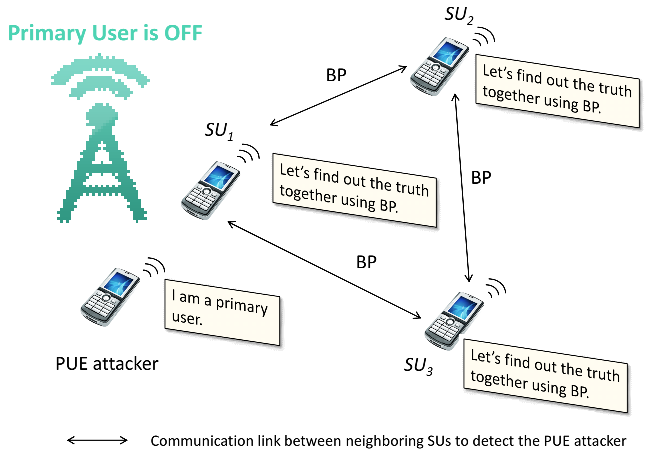
Primary User is OFF
SU_2
BP
Let's find out the truth together using BP.
SU_1
Let's find out the truth together using BP.
BP
BP
I am a primary user.
PUE attacker
SU_3
Let's find out the truth together using BP.
Communication link between neighboring SUs to detect the PUE attacker

Figure 2.1 Illustration of belief propagation based detection against a PUE attack in cognitive radio networks.

against the PUE attack in CR wireless networks. By comparing the distribution of the received signal power from the suspect and that from the primary user, each secondary user can have an approximate belief about the probability that whether a suspect is a PUE attacker or not, since the secondary user has no knowledge about the transmission output power of the attacker, as well as the distance from the attacker to the secondary user. In addition, the channel shadowing fading between each secondary user and the attacker may vary significantly. To accurately identify the attacker, a defense strategy based on belief propagation (BP) is developed. As shown in Figure 2.1, when the primary user is inactive, the PUE attacker will send primary user emulation signals to attack the cognitive radio network. When SUs receive this signal, they will perform local observations, and then use BP to exchange the information to detect whether the signal is from a PUE attacker or not. Each user will use the local functions to calculate the local estimation of the suspect, compute the compatibility functions to model the interactions between neighboring users, and update and exchange messages with the neighboring users in an iterative way using BP. After convergence, the PUE attacker can be detected according to the mean of all the final beliefs. If the mean of final belief values is lower than a threshold, the suspect can be detected as a PUE attacker. Otherwise, the suspect is seen as an honest secondary user. After that, all the secondary users in the network will be notified in a broadcast way about the PUE attacker's characteristics, and ignore the PUE attacker's primary emulation signal in the future. We also prove some properties of the proposed BP algorithm. Simulation results show that our proposed approach converges very fast, and is effective to detect the PUE attacker.

Compared to the other existing algorithms, our proposed approach has the following advantages:

- No additional cost is required in our approach for new hardware. We do not need to purchase wireless sensors and deploy an additional sensor network, which is required in [42]. Also in our proposed framework, different from [42], we do not need to calculate the exact location of the PUE suspect. Instead, we only need to exchange the beliefs between the neighboring users, and the attacker is identified by the final belief value. In addition, we can avoid the deployment of expensive hardware for TDOA and FDOA techniques, which are applied in [61].

- When there are many observations in each round of the detection, BP can automatically aggregate the observations into a much lower dimensional space of parameters. This is called the curse of dimension. In our proposed algorithm, only one parameter needs to be exchanged between neighboring users, while the direct observation exchange may incur substantial communication overhead.

- The performance of BP has been demonstrated by its success in many applications [67]. The elegant framework of BP is also flexible for modification and simplification. For example, the BP framework can be easily extended to detect various kinds of attacks beyond the PUE attack.

- In our proposed algorithm, BP converges very fast. From the simulation results, we can show that BP converges in a few iterations. In addition, we do not need to wait until the convergence of BP is reached. Specifically, the simulation shows that only a few iterations can provide good result compared to that after convergence is reached. This can significantly increase the efficiency of the proposed algorithm.

2.2 System Model and Simple Received Signal Based Detection Against PUE

In this section, we first describe the system model and then introduce a simple detection strategy based on the energy detection of the received signal power to determine whether such a received signal is from the PUE attacker or not.

To model the relationship between the transmit signal power and the received signal power, we consider the path loss and the log-normal shadowing of a communication channel[1]. The received signal from the primary user k can be defined as follows,

$$P_{r(PU_k)} = P_{t(PU_k)} d_{PU_k}^{-\alpha} h, \tag{2.1}$$

where d_{PU_k} is the distance between primary user k and the receiver, α is the path loss constant, $P_{t(PU_k)}$ represents the transmit power of primary user k, $P_{r(PU_k)}$ represents the received power from primary user k, and h is a shadowing random variable. Here we define $h = 10^{b/10} = e^{ab}$, where $a = \frac{\ln 10}{10}$ and b follows a normal distribution $b \sim N(0, \sigma^2)$. The moment generating function of normal random variable b is [68]

$$\varphi(t) = E\left(e^{tb}\right) = e^{\frac{1}{2}\sigma^2 t^2}, \tag{2.2}$$

which leads to

$$\begin{aligned} E(h) &= E(e^{ab}) = e^{\frac{1}{2}a^2\sigma^2}, & (2.3)\\ Var(h) &= E(h^2) - E^2(h) = e^{a^2\sigma^2}\left(e^{a^2\sigma^2} - 1\right). & (2.4) \end{aligned}$$

In a similar way, we can define the received signal power from the malicious node as

$$P_{r(attacker)} = P_{t(attacker)} d_{attacker}^{-\alpha} h_{attacker}, \tag{2.5}$$

where $d_{attacker}$ represents the distance from the attacker to the defender, although the defender does not know this distance. $P_{t(attacker)}$ and $P_{r(attacker)}$ represent the transmit power of the malicious node and the received power from the malicious node, respectively. A PUE attack is deemed

[1] The other types of fading such as Rayleigh fading can be implemented in a similar way.

successful if for a specified threshold ϵ, we have

$$|P_{r(PU_k)} - P_{r(attacker)}| < \epsilon, \tag{2.6}$$

and the probability of a successful PUE attack can be calculated as follows,

$$\mathcal{P}_{PUE} = Pr\{|P_{r(PU_k)} - P_{r(attacker)}| < \epsilon\}. \tag{2.7}$$

The defender receives the signal from primary user k, and the received signal strength in different time slots can be expressed as $P^1_{r(PU_k)}, P^2_{r(PU_k)}, \ldots, P^n_{r(PU_k)}, \ldots, P^N_{r(PU_k)}$, where n represents different time slots, and there are totally N time slots. We assume that $P^1_{r(PU_k)}$, $P^2_{r(PU_k)}$, $\ldots$, $P^n_{r(PU_k)}$, $\ldots$, $P^N_{r(PU_k)}$ are independent and identically distributed (i.i.d.) and follow the same distribution as $P_{r(PU_k)}$. Eq. (2.1) can be re-written as follows,

$$P_{r(PU_k)} = P_{t(PU_k)} d^{-\alpha}_{PU_k} h = P_{t(PU_k)} d^{-\alpha}_{PU_k} e^{ab}. \tag{2.8}$$

Therefore, the mean and the variance of $P_{r(PU_k)}$ can be calculated as follows,

$$\begin{aligned} u_{r(PU_k)} &= P_{t(PU_k)} d^{-\alpha}_{PU_k} e^{\frac{1}{2}a^2\sigma^2}, & (2.9) \\ \sigma^2_{r(PU_k)} &= P^2_{t(PU_k)} d^{-2\alpha}_{PU_k} e^{a^2\sigma^2} \left(e^{a^2\sigma^2} - 1\right), & (2.10) \end{aligned}$$

based on Eq. (2.3) and Eq. (2.4). Thus, we can use an unbiased estimator to estimate the mean and variance based on the measured data of $P^1_{r(PU_k)}, P^2_{r(PU_k)}, \ldots, P^n_{r(PU_k)}, \ldots, P^N_{r(PU_k)}$ by

$$\hat{u}_{r(PU_k)} = \frac{1}{N} \sum_{n_1=1}^{N} P^{n_1}_{r(PU_k)}, \tag{2.11}$$

$$\hat{\sigma}^2_{r(PU_k)} = \frac{1}{N-1} \sum_{n_2=1}^{N} \left(P^{n_2}_{r(PU_k)} - \frac{1}{N} \sum_{n_1=1}^{N} P^{n_1}_{r(PU_k)} \right)^2. \tag{2.12}$$

When N is large enough, we can get very accurate estimations of $u_{r(PU_k)}$ and $\sigma^2_{r(PU_k)}$. In this case, the defender can detect the attacker after comparing the characteristics of the received signal from the suspect and the received signal from the primary user.

However, the PUE attacker can use an advanced strategy to defeat the naive detection. To emulate the transmission of the primary user, the attacker can generate a signal that from the victim's point of view has the energy close to that from the primary user with the similar distribution. To achieve this, the PUE attacker first estimates the primary user transmit power and channel parameters. Then the attacker will perform the attack with the certain transmit power so that the received signal energy at the receiver side is as similar as possible to that from the primary user [62]. Therefore, we need to develop a more robust detection approach.

2.3 Detect PUE Attacker by Interactions Between Neighboring Users

In cognitive radio wireless networks, we assume that each secondary user is equipped with a localization unit (e.g. GPS), and has self-localization capability. Each secondary user is assumed to have a maximum transmission output power that is within the range from a few hundred milliwatts to a few watts, which corresponds to a typical cognitive radio transmission range. The primary users are assumed to be the TV signal transmitters (i.e., TV broadcast towers), and the locations of the primary users are fixed and known by the secondary users[2]. A TV tower's transmitter output power is typically hundreds of thousands of Watts [69], which corresponds to a transmission range from several miles to tens of miles. An attacker is a wireless node equipped with a cognitive radio functionality, which is capable of changing its modulation mode, frequency, location, and transmission output power. In the framework of BP, two secondary users can communicate with each other when their distance is within a threshold d_c. We are not discussing the detailed protocol of the communication between neighboring users, and simply assume that information can be exchanged without error. In practice, this can be accomplished by employing a reliable control channel.

Based on the above assumptions, we propose a transmitter verification scheme by calculating the location of the PUE attacker, and detecting the attacker by comparing the location of the attacker and the locations of the primary users. Since the primary users are placed at fixed locations, if a

[2] If the primary users are mobile users, secondary users can also get the information about the locations of the primary users based on the detection by the base stations.

signal source's estimated location deviates from the known locations of the TV towers, and the signal characteristics resemble those of the primary users, it is very likely that the signal source is a PUE attacker.

By the interactions between neighboring secondary nodes, received signal strength can be used to determine the location of the attacker. Assume that SU_1 and SU_2 are two secondary users, and their locations can be measured as (x_1, y_1) and (x_2, y_2) by using self equipped localization units, respectively. Since both secondary users SU_1 and SU_2 can receive the signal from the attacker, which is similar to the characteristics of the primary users' signal, both of them can record the RSSs of the attacker's signal. By the interactions between SU_1 and SU_2, SU_1 can obtain the full knowledge about the locations of both secondary users, and the RSSs of the attacker's signal at both locations of SU_1 and SU_2 — RSS_1 and RSS_2, respectively. If we re-write Eq. (2.1) to a statistical log-loss signal propagation model, which was shown to be appropriate for modeling signal propagation behavior in many situations [70], the RSS can be given by

$$P_r(\text{dBm}) = P_t(\text{dBm}) - \alpha \log(d) - s, \tag{2.13}$$

where P_r is the RSS in dBm, P_t is the transmission signal strength in dBm, d is the transmitter-receiver distance, α is again the path loss constant, and s represents the standard deviation associated with the degree of shadow fading. Since SU_1 and SU_2 are very close to each other, we can assume that $s_1 = s_2$ in the rest of this section for analysis purpose. In the next section, we investigate the case when $s_1 \neq s_2$.

Neither secondary users knows the transmit power of the attacker. Given the measured values of RSS_1 and RSS_2 and using Eq. (2.13), we can have

$$\eta = \frac{d_{attacker,SU_2}}{d_{attacker,SU_1}} = 10^{\frac{RSS_1 - RSS_2}{\alpha}} = 10^{\frac{\Delta RSS}{\alpha}}, \tag{2.14}$$

where $d_{attacker,SU_1}$ and $d_{attacker,SU_2}$ represent the distance from the attacker to SU_1 and SU_2, respectively.

Our objective is to calculate the location of the attacker without the knowledge of the transmit

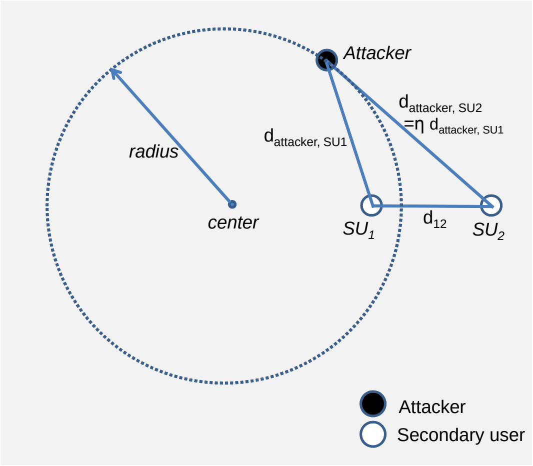

Figure 2.2 Illustration of the formation of the circle which represents the possible location of the PUE attacker by interactions between two neighboring users SU_1 and SU_2.

power of the attacker. Suppose that the location of the attacker is (x, y), using Eq. (2.14), we can have

$$\eta\sqrt{(x-x_1)^2+(y-y_1)^2}=\sqrt{(x-x_2)^2+(y-y_2)^2}. \tag{2.15}$$

Therefore, we can calculate the trace of the attacker, which is a circle with the radius, and the center axis

$$\text{radius}=\frac{\eta d_{12}}{\eta^2-1}, \text{center at}\left(\frac{\eta^2 x_1-x_2}{\eta^2-1},\frac{\eta^2 y_1-y_2}{\eta^2-1}\right), \eta\neq 1, \tag{2.16}$$

where d_{12} represents the distance between SU_1 and SU_2. Figure 2.2 illustrates the formation of the circle, which represents the possible location of the attacker by the interaction between two secondary users SU_1 and SU_2. Note that if $\eta=1$, the trace is a straight line orthogonal to the straight line (SU_1SU_2), passing through the midpoint of (SU_1SU_2), which makes the problem much easier to solve, since the intersection of such two straight lines is the location of the attacker. In here, we only consider the case that $\eta\neq 1$.

Figure 2.3 shows a simple simulation to illustrate how this works by assuming the transmission range is a radius. In Figure 2.3, four secondary users are deployed in a 700-by-700 meter area with the locations (400m, 300m), (600m, 300m), (350m, 480m), and (650m, 450m). SU_1 can generate a circle with the help of one of its neighboring users. One circle is not enough, and we need three circles to determine the exact location of the attacker. Therefore, one secondary user needs

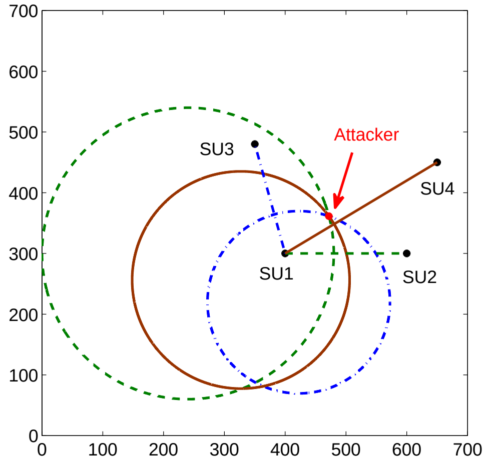

Figure 2.3 Simulation results of the proposed location detection strategies by interactions between neighboring users.

to interact with three neighboring nodes, and uses the intersection of the three circles to obtain the location of the attacker. By gathering the information of locations and RSSs from the neighboring nodes SU_2, SU_3, and SU_4, and by calculating the different values of ηs, secondary user SU_1 can plot three circles with the attacker at the intersection location (472m, 371m). Note that SU_2, SU_3, and SU_4 are not required to be in each other's transmission range, and they are just the neighboring users of SU_1. Then SU_1 will compare the location of the suspect and the locations of the primary users to determine whether the suspect is a PUE attacker or not.

2.4 Detect PUE Attacker Using Belief Propagation

In Section 2.3, we assume that in the idealized case, the standard deviation associated with the degree of shadow fading s in Eq. (2.13) remains the same between neighboring nodes. This is not realistic in practice, which may lead to the case that there is no common intersection point between three circles. Therefore, we need to develop new practical strategies.

BP can be used to provide a high accuracy of detection [71, 72]. In single-user decision making, only the local observation is used. To accurately detect the PUE attacker, all neighboring users can communicate with each other and exchange the information. We develop the BP framework based on Markov random field (MRF) [59]. In MRF, Y_v is defined as the quantities about the direct observations at node v, and X_v is the inferred quantities about the underlying information that is not obvious and need to be discovered. More specifically, in the process of defending the PUE attack, Y_v represents local observations at secondary user v, and X_v represents the state of the suspect observed by secondary user v. When $X_v = 0$, the suspect is a PUE attacker; otherwise, $X_v = 1$. We further define that there is some statistical dependency between X_v and Y_v at each SU_v, which we write as a local function $\phi_v(X_v, Y_v)$. The physical meaning of $\phi_v(X_v, Y_v)$ in our case is the relationship between the observations about the suspect's signals and the local decision of SU_v about whether the suspect is a PUE attacker or not. Finally, in Markov Random Fields, we also need to define a compatibility function $\psi_{vw}(X_v, X_w)$ to model the relationships between neighboring users. The higher the value of the compatibility function, the closer the observations between two neighboring SUs. Note that in our model, the Markov random field is pariwise because the compatibility function only depends on pairs of SU_w and SU_v, and the graphical model is undirected. Therefore, in the general framework of BP, the joint probability for the unknown variable X_v can be defined as follows,

$$\mathcal{P}(\{X_v\}, \{Y_v\}) = \prod_{v=1}^{V} \phi_v(X_v, Y_v) \prod_{v \neq w} \psi_{vw}(X_v, X_w). \tag{2.17}$$

Our goal is to compute the marginal probability, which can be mathematically defined as sums over all possible states of all other users. We call this marginal probability "belief," and denote the belief at user v by b_v. For small-scale networks, we can easily do marginalization sums directly. Nevertheless, when the number of the secondary users in the CR network increases, the number of terms in the sums will grow exponentially, which causes "the curse of dimension." This means that it is inefficient to do marginalization sums directly in large-scale networks. Therefore, we will use BP to compute marginal probabilities, which grows only linearly with the number of the users.

For the joint probability in Eq. (2.17), BP can be used to compute the marginal probability of user v iteratively. We introduce the variables $m_{wv}(X_v)$, which can be intuitively understood as a message from SU_w to SU_v. In the *l*-th iteration, secondary user w sends a message $m^l_{wv}(X_v)$ to secondary user v, which can be updated by

$$m_{wv}^{l}(X_v) = C_i \sum_{X_w} \psi_{wv}(X_w, X_v) \phi_w(X_w, Y_w) \prod_{u \neq w,v} m_{uw}^{l-1}(X_w), \tag{2.18}$$

where C_i is a normalized factor such that $m_{wv}^{l}(1) + m_{wv}^{l}(0) = 1$.

When all iterations are completed, secondary user v computes its belief using

$$b_v(X_v) = k_v \phi_v(X_v, Y_v) \prod_{w \neq v} m_{wv}(X_v), \tag{2.19}$$

where k_v is a normalization constant. We can see that in Eq. (3.12), the belief at a secondary user is proportional to the product of the local evidence at this user and all the messages coming into this user. This means that BP is a way of organizing the "global" computation of marginal beliefs in terms of smaller local computations. The message flow is similar to the flow of a river, and each message summarizes all the computations that occurred further upstream along the branches that feed into that message. Also, it is easy to see that each message only needs to be computed only once, which means that the whole computation takes an amount of time proportional to the number of links in the graph, which is dramatically less than the exponentially large time that would be required to compute marginal probabilities naively [59]. In addition, in our framework, only one parameter needs to be exchanged between neighboring users, which makes the communication overhead very low, while the direct observation exchange may incur substantial communication overhead.

The convergence of BP can be proved using the following theorem.

Theorem 2.1. *BP converges to a unique fixed point, irrespective of the initial messages, if the spectral radius of the $|D| \times |D|$-matrix*

$$O_{i \to j, k \to l} := \tanh |J_{ij}| \delta_{i,l} \mathbf{1}_{\partial i \setminus j}(k) \tag{2.20}$$

is strictly smaller than 1.

Proof.

Theorem 2.2. *Let $f : X \to X$ be a contraction of a complete metric space (X, d). Then f has a unique fixed point $x_\infty \in X$ and for any $x \in X$, the sequence x, $f(x)$, $f^2(x)$, $\ldots$ obtained by iterating f converges to x_∞. The rate of convergence is at least linear, since $d(f(x), x_\infty) \leq Kd(x, x_\infty)$ for all $x \in X$ [73].*

Lemma 2.3. *Let $(V, ||\cdot||)$ be a normed space and $f : V \to V$ a differentiable mapping. Then for $x, y \in V$*

$$||f(y) - f(x)|| \leq ||y - x|| \cdot \sup_{z \in [x,y]} ||f'(z)||, \tag{2.21}$$

where we wrote $[x, y]$ for the segment $\{\lambda x + (1 - \lambda)y : \lambda \in [0, 1]\}$ joining x and y [74].

Combining Theorem 2.2 and Lemma 2.3, we can have

Lemma 2.4. *Let $(V, ||\cdot||)$ be a normed space, $f : V \to V$ differentiable and suppose that*

$$\sup_{v \in V} ||f'(v)|| < 1. \tag{2.22}$$

Then f is a $||\cdot||$-contraction by Lemma 2.3. Therefore, for any $v \in V$, the sequence v, $f(v)$, $f^2(v)$, $\ldots$ converges to a unique fixed point $v_\infty \in V$ with a convergence rate that is at least linear by Theorem 2.2.

The BP update equation can be re-written as follows [73],

$$\tanh \tilde{v}^{i\to j} = \tanh(J_{ij}) \tanh\left(\theta_i + \sum_{t \in \partial i \setminus j} v^{t \to i}\right), \tag{2.23}$$

where $\partial i = \{t \in \mathcal{N} : \{i, t\} \in \Upsilon_2\}$ are the variables that interact with i via a pair-potential. If we apply Lemma 2.4 to Eq. (2.23), the derivative of f is given by

$$(f'(v))_{i\to j,k\to l} = \frac{\partial \tilde{v}^{i\to j}}{\partial \tilde{v}^{k\to l}} = O_{i\to j,k\to l} R_{i\to j}(v), \tag{2.24}$$

where

$$R_{i\to j}(v) \quad := \quad \frac{1 - \tanh^2(\theta_i + \sum_{t \in \partial i \setminus j} v^{t\to i})}{1 - \tanh^2(\tilde{v}^{i\to j}(v))} \mathrm{sgn} J_{ij}, \tag{2.25}$$

$$O_{i\to j,k\to l} \quad := \quad \tanh|J_{ij}| \delta_{i,l} \mathbf{1}_{\partial i \setminus j}(k). \tag{2.26}$$

Instead of pursuing a search for the optimal norm, we will derive a criterion for convergence based on the spectral radius of the matrix in Eq. (2.26).

Lemma 2.5. *Let $f : X \to X$ be a mapping, d a metric on X and suppose that f^N is a d-contraction for some $N \in \mathbb{N}$. Then f has a unique fixed point x_∞ and for any $x \in X$, the sequence x, $f(x)$, $f^2(x)$, $\ldots$ obtained by iterating f converges to x_∞.*

Theorem 2.6. *Let $f : \mathbb{R}^m \to \mathbb{R}$ be differentiable and suppose that $f'(x) = R(x)O$, where O has nonnegative entries and R is diagonal with bounded entries $|R_{ii}(x)| \leq 1$. If $\rho(O) < 1$, then for any $x \in \mathbb{R}^m$, the sequence x, $f(x)$, $f^2(x)$, $\ldots$ obtained by iterating f converges to a fixed point x_∞, which does not depend on x.*

Therefore, using Eq. (2.24), Eq. (2.25) and Eq. (2.26), we can yield Theorem 2.1. □

After all the SUs finish computing their final beliefs, if the mean of all beliefs for the suspect is lower than a threshold b_τ, the suspect user can be seen as a PUE attacker. Therefore, the decision rule is

$$\text{Suspect} = \begin{cases} \text{honest}, & \frac{1}{M}\sum_v b_v \geq b_\tau; \\ \text{malicious}, & \frac{1}{M}\sum_v b_v < b_\tau, \end{cases} \tag{2.27}$$

where M represents the total number of secondary users in the CR network.

Note that there may exist malicious SUs who can lie to their neighbors. However, the proposed BP-based detection scheme can avoid this problem, and actually this is a major advantage of the proposed scheme. In the iterations of BP, the information is exchanged between neighboring users. Although some of the information is inaccurate or fake because of the existence of the malicious users, the final belief after convergence can still be used to detect the PUE attacker, because most of the messages are accurate except those from the malicious SUs. On the other hand, if most of the SUs behave dishonestly, there will be problem for BP-based detection since the final belief value after convergence can be inaccurate. However, this is a cross-layer attack, which is beyond the coverage of this paper.

2.4.1 Local Function

In belief propagation, we need to define the local function $\phi_v(X_v, Y_v)$ and the compatibility function $\psi_{wv}(X_w, X_v)$ to formulate the defense problem. In this subsection, we focus on the local function. We first examine the received signal from the primary users. Based on Eq. (2.1), for two neighboring SUs, we can have

$$\frac{P_{r1(PU_k)}}{P_{r2(PU_k)}} = \left(\frac{d_{1(PU_k)}}{d_{2(PU_k)}}\right)^{-\alpha} \left(\frac{h_{1(PU_k)}}{h_{2(PU_k)}}\right), \tag{2.28}$$

where PU_k represents the *k*th primary user, $P_{r1(PU_k)}$ and $P_{r2(PU_k)}$ are received PU_k's signal strength at SU_1 and SU_2, and $d_{1(PU_k)}$ and $d_{2(PU_k)}$ are the distance between SU_1, SU_2 and PU_k. $h_{1(PU_k)}$ and $h_{2(PU_k)}$ represent two independent shadowing channels between SU_1, SU_2 and PU_k, respectively. We assume that the channel response is a circular Gaussian random variable $\mathcal{C}N(0,1)$. Therefore, if we simplify $h_{1(PU_k)}$ and $h_{2(PU_k)}$ to h_1 and h_2, we can obtain the probability density function (PDF) of $q = \frac{h_{1(PU_k)}}{h_{2(PU_k)}}$ as follows,

$$\begin{aligned} f_q(q) &= \int_{-\infty}^{\infty} |h_2| f_{h_1 h_2}(q h_2, h_2) d(h_2) \\ &= \int_0^{\infty} q |h_2| {h_2}^2 e^{-\frac{1}{2}(q^2 h_2^2 + h_2^2)} d(h_2) \\ &= \frac{2q}{(q^2+1)^2}. \end{aligned} \tag{2.29}$$

In this way, if we define $B = \left(\frac{d_{1(PU_k)}}{d_{2(PU_k)}}\right)^{-\alpha}$, we can obtain the distribution of $\frac{P_{r1(PU_k)}}{P_{r2(PU_k)}}$ as follows,

$$PDF_{PU_k} = PDF_{\frac{P_{r1(PU_k)}}{P_{r2(PU_k)}}} = \frac{1}{|B|} \frac{2\frac{q}{B}}{((\frac{q}{B})^2+1)^2}. \tag{2.30}$$

Note that since the secondary users know the exact locations of the primary users, the value of B, $d_{1(PU_k)}$ and $d_{2(PU_k)}$ are known to SU_1. In addition, from Eq. (2.29), we can also calculate the expectation of q as follows,

$$E(q) = \int_{-\infty}^{\infty} q f_q(q) d(q) = \int_0^{\infty} q \frac{2q}{(q^2+1)^2} d(q) = \pi. \tag{2.31}$$

To detect the PUE attacker, a secondary user needs to collect the information from one of its neighboring nodes, which includes the location of this neighboring node, as well as the received signal strength of the attacker's signal. If we define the received signal strength of the attacker's signal at each user as $P_{r1(attacker)}$ and $P_{r2(attacker)}$, and the distance between SU_1, SU_2 and the attacker as $d_{1(attacker)}$ and $d_{2(attacker)}$, we can obtain the estimated value of $\left(\frac{d_{1(attacker)}}{d_{2(attacker)}}\right)^{-\alpha}$ as follows,

$$A = \left(\frac{d_{1(attacker)}}{d_{2(attacker)}}\right)^{-\alpha} = \frac{P_{r1(attacker)}/P_{r2(attacker)}}{\pi}, \tag{2.32}$$

where π is the expectation in Eq. (2.29). Therefore, the distribution of $\frac{P_{r1(attacker)}}{P_{r2(attacker)}}$ can be written as follows,

$$PDF_{attacker} = PDF_{\frac{P_{r_1(attacker)}}{P_{r_2(attacker)}}} = \frac{1}{|A|}\frac{2\frac{q}{A}}{((\frac{q}{A})^2+1)^2}. \tag{2.33}$$

Kullback Leibler (KL) distance can be used to obtain the difference between two distributions [75]. Based on the definition of KL distance, for the two distributions PDF_{PU_k} and $PDF_{attacker}$, the Kullback Leibler distance can be calculated as follows,

$$KL_{(PDF_{PU_k}, PDF_{attacker})} = \int_0^\infty PDF_{PU_k} \log \frac{PDF_{PU_k}}{PDF_{attacker}} d(q). \tag{2.34}$$

Therefore, the local function can be computed as the exponential function of the KL distance, which is always between 0 and 1 as follows,

$$\phi_v = \exp\left(-\min_k KL_{(PDF_{PU_k}, PDF_{attacker})}\right). \tag{2.35}$$

The physical meaning of the local function ϕ_v is the probability that the suspect is a primary user based on the observation of secondary user v. When the value of KL distance is high, which means large difference between the two distributions, we can obtain a low value of ϕ_v, which represents a high probability that the suspect is a PUE attacker.

2.4.2 Compatibility Function

We only consider the shadowing fading. It is prohibitively difficult to find a closed form expression for the compatibility function $\psi_{wv}(X_w, X_v)$ in the case of shadowing fading. We notice that $\psi_{wv}(X_w, X_v)$ is dependent on the correlation between the states X_w and X_v. If X_w and X_v are highly positively correlated, i.e., they tend to be the same, $\psi_{wv}(X_w, X_v)$ should yield a large probability for $X_w = X_v$ and a small probability for $X_w \neq X_v$. Therefore, we propose to use the following exponential compatibility function, which can be given by

$$\psi_{wv}(X_w, X_v) = \exp(-Cd^{\beta}_{X_w,X_v}), \tag{2.36}$$

where C and β are two constants, and d_{X_w,X_v} represents the distance between secondary user w and secondary user v. We can find that this compatibility function depends on the distance between two SUs. When the distance is large, the value of compatibility function is low. The exponential function guarantees that the compatibility value is always between 0 and 1. Also the proposed compatibility function in Eq. (3.24) is symmetric for both random variables X_w and X_v, which satisfies our requirements. We consider the shadowing fading, and in Section 2.5, we will provide simulation results to prove that the proposed exponential compatibility function works well in the shadowing fading case.

Note that the proposed exponential compatibility function can be used for other scenarios, such as additive white Gaussian noise (AWGN) and Rayleigh fading. Although we only consider the shadowing fading case, we provide the formulations of the compatibility functions for other scenarios in Appendix 2A at the end of this chapter.

2.4.3 Complete Algorithm

The complete belief propagation algorithm is summarized in Algorithm 3.1. All the secondary users will first perform the measurements to calculate the distributions of PDF_{PU_k} and $PDF_{attacker}$ using Eq. (2.30) and Eq. (2.33). Then in an iterative way, each secondary user calculates the local function and the compatibility function, computes the messages, exchanges messages

Algorithm 2.1 Complete defense strategy against the PUE attack using belief propagation

```
Each secondary user performs measurements using Eq. (2.30) and Eq. (2.33).
for Each iteration do
    Compute the local function using Eq. (2.35), and the compatibility function using Eq. (3.24).
    Compute messages using Eq. (3.11).
    Exchange messages with neighbors.
    Compute beliefs using Eq. (3.12).
end for
The PUE attacker is detected according to the mean of all final beliefs based on Eq. (2.27).
Each secondary user will be notified about the characteristics of the attacker's signal, and ig-
nores the attackers primary emulation signal in the future.
```

with neighboring users, and calculates the beliefs until convergence. In the next step, the PUE attacker will be detected according to the mean of the final beliefs based on the decision rule in Eq. (2.27). If the mean of the final belief values is higher than a threshold b_{τ}, the suspect can be seen as a primary user. Otherwise, a PUE attacker is detected. After that, all the secondary users in the network will be notified in a broadcast way about the characteristics of the attacker's signal. In the future, all the secondary users will ignore the PUE attacker's primary emulation signal.

Note that a PUE attacker can choose its location and power to make some other SU nodes think that it is a PU. However, there are many SUs deployed in a wide area in the network, and only a small number of the SUs that are certain distance away from the PUE attacker can be deceived by the PUE attacker. Many SUs are able to detect the suspicious behavior of the PUE attacker, and by exchanging messages using belief propagation, the PUE attacker can be identified, even though some of the messages from those deceived SUs are not accurate. Therefore, in our proposed BP-based detection algorithm, we are able to detect the PUE attacker regardless of the location and power it chooses.

2.5 Simulation Results and Analysis

In this section, we use numerical simulations to demonstrate the performance of the BP-based detection against PUE attack in CR networks. In the simulations, the path loss constant α is set to be 2.5, constant β is also set to be 2.5 and the channels are defined as shadowing variables. The transmit power of the secondary users is set to be 0.1 W. In Figure 2.4, there are 30 secondary users randomly

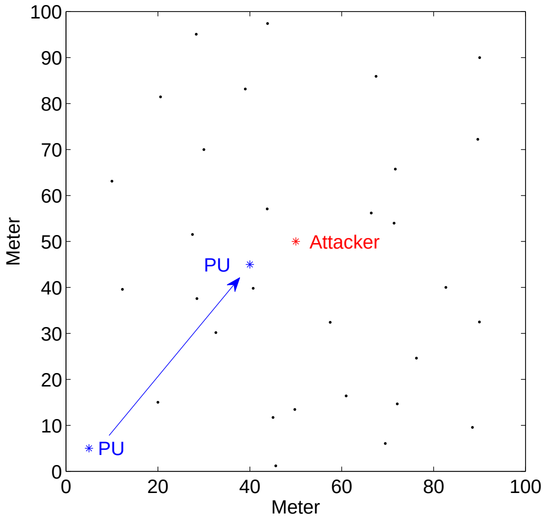

Figure 2.4 Illustration of the CR network model.

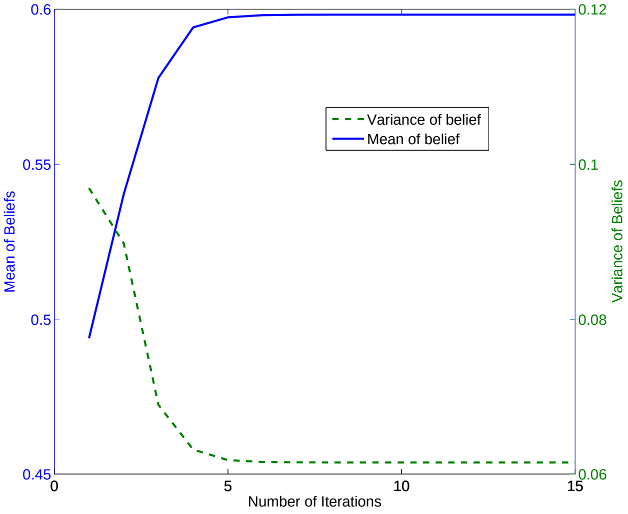

Figure 2.5 Convergence of belief propagation when the primary user is at (5m, 5m).

deployed in a 100 meter-by-100 meter area, and we assume that the maximum transmission range of each secondary user d_c is 20m. The PUE attacker is at the location of (50m, 50m), and we assume that there is only one primary user in the network. The transmission power of the PUE attacker is 0.1W. When the primary user's location is at (5m, 5m), the simulation results are shown in Figure 2.5. The solid line represents the mean of the beliefs over time, and the dashed line is the variance of the beliefs over time. We can find that BP converges very fast (converges in 5 iterations), and

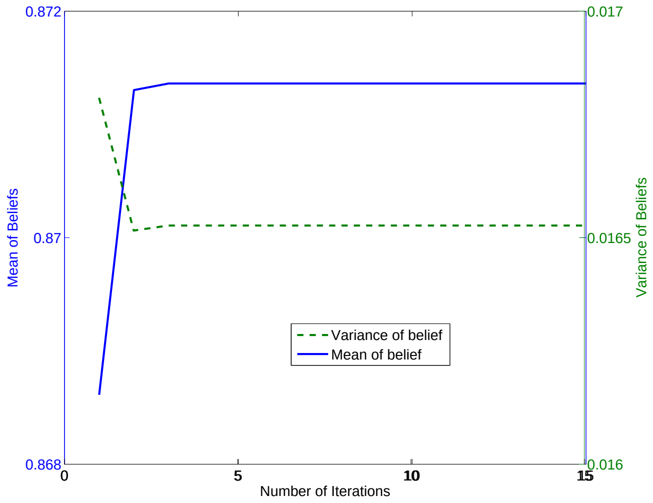

Figure 2.6 Convergence of belief propagation when the primary user is at (40m, 45m).

the mean of final belief values is around 0.6. When the primary user's location is at (40m, 45m) instead of (5m, 5m) in the same network, the simulation results are shown in Figure 2.6. In this case, the mean of the final belief values is around 0.871, which finishes convergence in 3 iterations. After comparing the results in Figure 2.6 and Figure 2.5, we can find that the final belief is higher in Figure 2.6 than that in Figure 2.5. In other words, when the distance between the primary user and the suspect is smaller, the belief will be higher, because there is a higher probability that the suspect is a primary user. In addition, the variance of belief in Figure 2.5 is 0.062, which is higher than 0.016 shown as in Figure 2.6. This shows higher uncertainty of beliefs and higher probability that the suspect is a PUE attacker when the distance between the primary user and the attacker is larger.

Figure 2.7 provides the simulated relationship between the distance between the PUE attacker and the primary user and the final belief values. The simulation results are based on 1000 different scenarios in which 30 secondary users are randomly deployed in a 100m-by-100m area, not restricted to the deployment in Figure 2.4. From the results in Figure 2.7, we can clearly see that when the distance between the primary user and the suspect is bigger, the mean of the final belief values is lower, which represents a higher probability of being a PUE attacker.

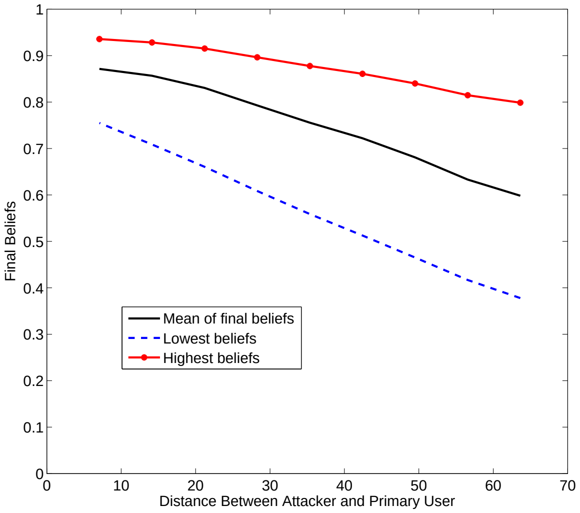

Figure 2.7 Distance between the primary user and the PUE attacker vs. the final belief values.

Table 2.1 Number of the secondary users in the CR network vs. the number of iterations to converge.

Number of SUs	Number of iterations to converge
30	7.8943
40	7.6041
50	7.4790
60	7.8625
70	7.7004

Table 2.1 shows the simulation results regarding the number of the secondary users in the CR network vs the number of iterations for BP to converge. Again, different number of secondary users are randomly deployed in a 100m-by-100m area, and the simulation is based on the average results of 1000 scenarios. We can find that in our simulations the number of iterations to converge remains below 8, and the number of secondary users does not actually affect the number of iterations for BP to converge. This again proves that the BP-based defense strategy converges very fast, and it can be employed in large-scale CR networks.

Figure 2.8 shows the probability of missing detection and the probability of false alarm when the belief threshold b_T changes. We can observe that when the belief threshold b_T increases, missing detection probability decreases, and the false alarm probability increases.

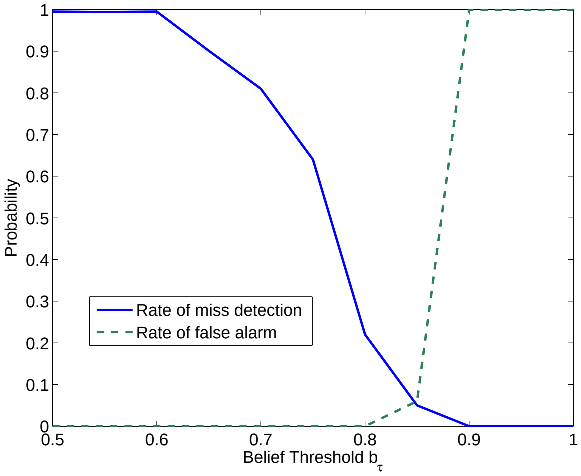

Figure 2.8 Missing detection probability and false alarm probability over different belief thresholds.

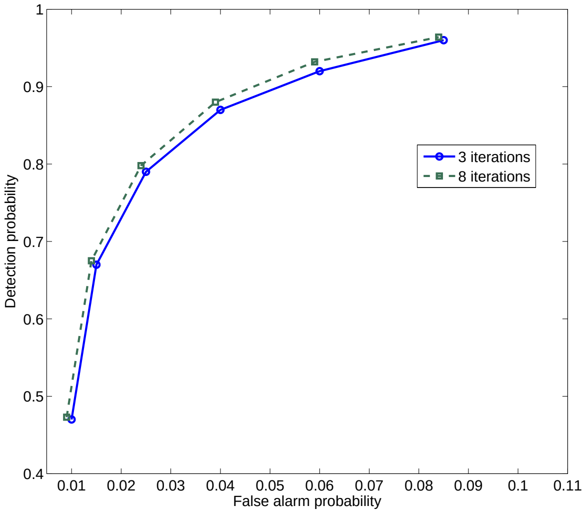

Figure 2.9 Probability of false alarm vs. successful detection probability for different iterations of belief propagation.

In Figure 2.9, the comparison between the results after 3 iterations of belief propagation and that after 8 iterations of belief propagation is provided. The reason that we use the results after 8 iterations is that in Table 2.1, we have proved that in our setup, BP can converge in 8 iterations, no matter how many secondary users are deployed in this area. In this simulation b_τ is set to be 0.85. We can see that when false alarm probability increases, detection probability increases too, although the amount of increment becomes smaller and smaller. We can also observe in Figure 2.9 that the

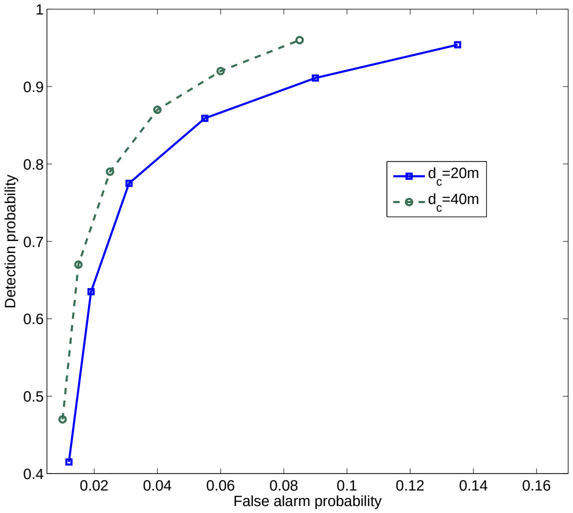

Figure 2.10 Probability of false alarm vs. successful detection probability given different values of secondary communication range d_c.

two curves are very close to each other. This implies that more iterations may not lead to better performance, and 3 iterations of belief propagation can already provide good enough solutions in our problem. Therefore, we can spend only a few iterations to obtain a good result instead of waiting until the BP algorithm to completely converge.

Figure 2.10 shows the simulation comparison for different values of secondary communication range d_c. We know that when the secondary communication range is larger, there is more collaborations between SUs, since each SU can be connected to more neighbors. Therefore, we can observe in Figure 2.10 that there is a significant performance improvement when the communication range is increased from 20m to 40m.

2.6 Conclusion

In this chapter, we have proposed a belief propagation-based defense strategy against PUE attack in cognitive radio wireless networks. In our proposed approach, each secondary user calculates the local functions and compatibility functions, computes the messages, exchanges messages with the neighboring users, and calculates the beliefs until convergence. Then, the PUE attacker will be detected according to the mean of the final beliefs based on a belief threshold. If the mean of the final beliefs is below a threshold, the suspect can be detected as a PUE attacker. Finally, all the secondary users in the network will be notified about the characteristics of the attacker's signal, and avoid the attacker's primary emulation signal in the future. Our proposed framework can avoid deploying the expensive hardware, and is flexible to incorporate various kinds of attacks for future extension. Simulation results show that our proposed technique converges very fast, even in large-scale networks. Also a few iterations of belief propagation can provide very good results, and it is very effective and efficient to detect the PUE attacker.

Appendix 2A: Compatibility Functions for AWGN and Rayleigh Fading

In Section 2.4.2, we formulate the compatibility function for the shadowing case. In this appendix, we show that our model can be easily expanded into AWGN channel and Rayleigh fading scenarios.

AWGN Channel

In the case of AWGN channel, if we assume BPSK modulation for the communications between neighboring SUs, the bit error rate (BER) can be calculated using the following Q-function [76],

$$BER = Q(\sqrt{2\gamma_{awgn}}), \tag{2.37}$$

where γ_{awgn} represents the signal to noise ratio (SNR).

When two SUs are closer to each other, the communications between them will have lower BER, which contributes to a higher compatibility function. Therefore, the compatibility function for the AWGN case can be defined as,

$$\psi_{wv}(X_w, X_v) = 1 - Q(\sqrt{2\gamma_{awgn}}) = 1 - Q\left(\sqrt{2\frac{P_{t,SU} d_{X_w,X_v} h_{X_w,X_v}}{N_0}}\right). \tag{2.38}$$

Rayleigh Fading

If the communication channel between two secondary users SU_1 and SU_2 is Rayleigh fading, the probability density function (PDF) of the SNR about the received signal from SU_1 to SU_2 can be expressed as follows,

$$\mathcal{P}(\gamma_R) = \frac{N_0 d^{\alpha}_{SU_1,SU_2}}{P_{t,SU}} e^{-\frac{\gamma_R N_0 d^{\alpha}_{SU_1,SU_2}}{P_{t,SU}}}, \gamma_R > 0, \tag{2.39}$$

where γ_R represents the SNR, N_0 is the background noise power, and d_{SU_1,SU_2} is the distance between SU_1 and SU_2. $P_{t,SU}$ is the transmit power of a secondary user, and we assume that the transmit power of each secondary user is the same. The probability that the signal can be successfully received can be calculated by

$$\int_{\Gamma_{\gamma_R}}^{\infty} \mathcal{P}(\gamma_R) d\gamma_R = e^{-\frac{\Gamma_{\gamma_R} N_0 d^{\alpha}_{SU_1,SU_2}}{P_{t,SU}}}, \tag{2.40}$$

where Γ_{γ_R} is the required SNR such that the signal can be detected by the secondary user at the receiver side.

Therefore, we can use the following compatibility function

$$\psi_{wv}(X_w, X_v) = \exp\left(-\frac{\Gamma_{\gamma_R} N_0 d^{\alpha}_{X_w,X_v}}{P_{t,SU}}\right). \tag{2.41}$$

Chapter 3

Routing-Toward-Primary-User Attack and Belief Propagation Based Defense in Cognitive Radio Networks

In this chapter, we first introduce the concept of routing-toward primary user (RPU) attack in Section 3.1. Then, Section 3.2 states the RPU attack model and the problem formulation. In an RPU attack, malicious nodes intentionally route a large amount of packets toward the primary users, aiming to cause interference to the primary users and to increase delay in the data transmission among the secondary users. In the RPU attack, it is difficult to detect the malicious nodes since the malicious nodes may claim that those nodes, to which they forward the packets, behave dishonestly and cause problems in the data transmission. In Section 3.3 the defense strategy based on belief propagation is developed, in which each node keeps a table recording the feedbacks from the other nodes on the route, exchanges feedback information, computes beliefs and detects the malicious nodes based on the final belief values. Then simulation results are given in Section 3.4 which show that the proposed defense strategy against the RPU attack is effective and efficient in terms of significant reduction in the delay and interference caused by the RPU attack. Finally, in Section 3.5, we conclude the chapter.

3.1 Routing-Toward-Primary-User (RPU) Attack

Cognitive radio (CR) is a revolutionary technique that allows secondary user (SU) wireless devices to use spectrum holes left by idle primary users (PUs) to improve the wireless spectrum usage. In CR wireless networks, wireless routers work as secondary users that can opportunistically utilize various spectral holes for communications without causing interference to the primary users. For distributed cognitive radio networks, the network is dynamically self-organized and self-configured, with the nodes in the network automatically establishing and maintaining connectivity, as shown in some recent work [16–20, 77–80].

Although CR wireless networks have so many advantages, most CR networks do not offer sufficient security. Many of the security challenges are due to the fact that the networks inherently rely on cooperation among distributed entities, such as collaborative sensing and multi-hop routing. Cooperation can be fragile under malicious attacks. Many attacks have been discovered for cognitive radio based networks, which has been discussed in Chapter 1, including common control channel (CCC) attack [81], primary user emulation (PUE) attack [42], reporting false sensing data (RFSD) attack [47], reporting false selection frame (FSF) attack [49], and false evacuation (FE) attack [50].

In this chapter, we introduce a new and powerful network layer attack in cognitive radio networks: routing-toward-primary-user (RPU) attack. In the RPU attack, the malicious nodes intentionally route a large amount of packets toward or around the primary users, aiming to cause interference to the primary users and to increase delay in data transmission along the CR routes. In this attack, even if the interference from a single CR device to the primary users is not severe, the aggregate effects from many cognitive radio devices transmitting at the same time around the primary users can be significant, which can largely impair the primary users' performance. Furthermore, the interference to the primary users is not directly generated by the malicious nodes. Instead, the interference is from the honest nodes that received the packets from the malicious nodes. In other words, malicious nodes "neutralize the route by the hands of the others." Therefore, it is difficult to detect the malicious nodes.

In the following sections, besides raising the awareness of the RPU attack and demonstrating its damage, we develop a defense strategy based on belief propagation (BP) against the RPU attack. Firstly, an initial route will be found from the source to the destination without knowledge of the malicious nodes. Each node on the route will keep a table recording the feedback information from the other nodes on the route. Then, every node exchanges the feedbacks with its neighboring nodes and computes beliefs. After this belief propagation converges, the source node can detect the malicious nodes based on the final belief values. Finally, the source node will re-route data packets to the destination, avoiding the detected malicious nodes. The underlying reason for applying BP

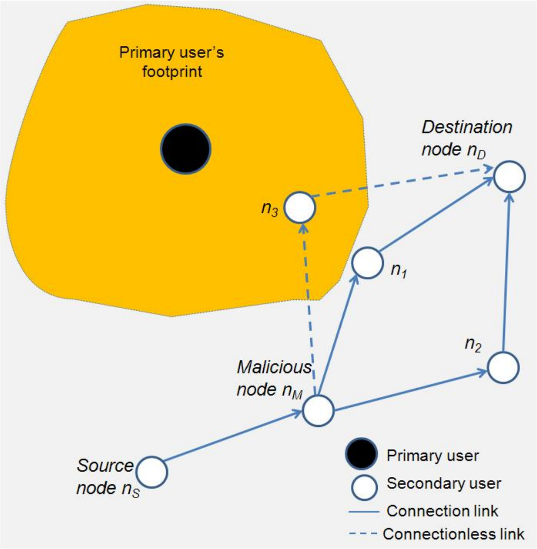

Figure 3.1 Illustration of the RPU attack.

is to reduce the complexity for such defense mechanism. Simulation results show that the proposed scheme is efficient and effective to detect the RPU attackers.

3.2 RPU Attack Model and Problem Formulation

3.2.1 RPU Attack Model in Cognitive Radio Networks

In the RPU attack, malicious nodes send fake routing information, claiming that they have optimum route with low costs, which will cause other honest nodes to route data packets through those malicious nodes. For example, using a simple shortest path routing algorithm[1] and assuming that there is only one malicious node, this malicious node may claim that the cost between itself and a secondary user, which is very close to the primary user, is very low. In this way, the honest nodes will forward data packets to this malicious node and all traffic will be routed through the attacker.

As shown in Figure 3.1, n_S, n_D, n_1, n_2 and n_3 are secondary users, and n_S and n_D represent the source and destination nodes, respectively. The shaded region is the footprint of a primary user. Since secondary node n_3 is inside this region, it is forced to be turned off at a specific time slot. Note

[1] The RPU attack is not limited for the shortest path routing algorithm, but also applied for other routing algorithms, like QoS routing, etc. In any routing algorithm, the malicious node can suggest to forward the packets to a node that is closer to the primary users to reduce QoS.

that because of channel fading, the primary users' footprint can change and have different shapes at different time slots. Therefore, each secondary node has a probability to be out of the primary user's footprint. If its distance to the primary user is shorter, it will have a higher probability to be inside the primary user's footprint and be turned off. In Figure 3.1, source node n_S wants to transmit some packets to destination node n_D, since the malicious node n_M claims that it has the shortest path to destination node n_D, source node n_S will forward all the packets to node n_M. Then node n_M tries to forward the data to node n_1, which is closer to the primary user compared to node n_2, even though node n_M knows clearly that node n_2 can also help to forward the packets. This has two consequences. First, the interference to the primary user along the route $n_M \rightarrow n_1 \rightarrow n_D$ is higher than the route $n_M \rightarrow n_2 \rightarrow n_D$. Second, since node n_1 is closer to the primary user and has a higher probability to be inside the primary user's footprint, it may be turned off more frequently and a larger transmission delay may occur. In the RPU attack, the malicious node itself can be at any location and it does not need to be close to the primary users. In other words, the malicious nodes are not the nodes that directly cause interference to the primary users or introduce long delay in data transmission. As a consequence, the source node cannot easily identify which node is a bad guy since the malicious node can argue that those nodes, to which it forwards the packets (i.e. node n_1 in this example), behave dishonestly and cause problems in data transmission.

We need to emphasize that the RPU attack is different from many known attacks in the literature. The RPU attack can not only cause data transmission failure, but also degrade the primary users' performance. Moreover, in the proposed RPU attack, instead of causing problems to the networks by the malicious node itself, the malicious node makes the honest nodes unintentionally hurt the network. As a consequence, the malicious nodes can be pretty covert and difficult to detect.

3.2.2 Channel Model

In cognitive radio wireless networks, secondary users need to sense the spectrum to see whether the spectrum is occupied by the primary users or not to avoid the interference with the primary users' transmissions. When a primary user $k_j \in \mathcal{K}$ is transmitting, the received signal at secondary user $n_i \in \mathcal{N}$ can be modeled as,

$$y_{n_i} = \sqrt{P_{t,k_j} K d_{n_i,k_j}^{-\alpha_{pl}}} h_{n_i,k_j} x_{n_i,k_j} + w, \tag{3.1}$$

where P_{t,k_j} is the transmitted signal power from primary user k_j, K is a constant that depends on the antennas design, d_{n_i,k_j} is the distance between secondary user n_i and primary user k_j, α_{pl} is the path loss constant, x_{n_i,k_j} is the transmitted data with unit power, h_{n_i,k_j} is the channel fading gain, and w represents the noise. The channel fading of any link is modeled as an independent zero mean circularly symmetric complex Gaussian random variable with unit variance. Therefore, the received primary user k_j's signal power at secondary user n_i can be defined as

$$P(d_{n_i,k_j})_{n_i} = |h_{n_i,k_j}|^2 K d_{n_i,k_j}^{-\alpha_{pl}} P_{t,k_j}. \tag{3.2}$$

We consider the system performance in terms of the probability that the received power at secondary user n_i from primary user k_j falls below a certain threshold P_τ, which means the spectrum is available for secondary user n_i. Therefore, n_i can be turned on and communicate with other secondary users in the CR networks causing negligible errors in primary user k_j's transmission. The power threshold P_τ is determined by secondary user n_i according to application scenarios and the transmitter/receiver structure. Only when the received primary user's power is lower than this threshold, could the second users transmit. The probability that secondary user n_i can be turned on is defined as

$$\mathcal{P}_{on,n_i} = \mathcal{P}\left[P(d_{n_i,k_j})_{n_i} \leq P_\tau, \forall_{k_j}\right], \tag{3.3}$$

in which we need to consider the received power from any primary user $k_j \in \mathcal{K}$. Therefore, the probability that secondary user n_i can be turned on is calculated as [82]

$$\mathcal{P}_{on,n_i} = \prod_{k_j \in \mathcal{K}} \left[1 - \exp\left(-\frac{P_\tau d_{n_i,k_j}^{\alpha_{pl}}}{K P_{t,k_j}}\right)\right]. \tag{3.4}$$

From (3.4), we can see that when the distance between a secondary user and a primary user increases, the probability that this secondary user can be turned on also increases, since it has a lower probability to interfere with primary users' transmissions.

3.2.3 Routing Cost for Secondary User

The shortest path routing algorithm can be used for routing among secondary users in CR wireless networks [83], which has been proved to be effective and efficient. The cost of each direct link can be defined considering the delay, which is inversely proportional to the capacity. In addition, we need to consider the availability of each link. When the capacity of the link is higher and the probability of the secondary nodes along the link being available is higher, the cost of the link is lower. Considering a geometric distribution, the probability that secondary user n_i is not available until the m^{th} trial is $(1-\mathcal{P}_{n_i})^m\mathcal{P}_{n_i}$, and the expectation is calculated as $\frac{1-\mathcal{P}_{n_i}}{\mathcal{P}_{n_i}}$, where $\mathcal{P}_{n_i}$ represents the probability of availability of secondary user n_i on each trial. Therefore, the cost $\mathcal{C}_{n_{i_1},n_{i_2}}$ for edge (direct link) $e_{n_{i_1},n_{i_2}}$ can be defined as

$$\mathcal{C}_{n_{i_1},n_{i_2}} = \frac{1}{c_{n_{i_1},n_{i_2}}}\frac{1-\mathcal{P}_{n_{i_1}}}{\mathcal{P}_{n_{i_1}}}, \tag{3.5}$$

where $c_{n_{i_1},n_{i_2}}$ is edge $e_{n_{i_1},n_{i_2}}$'s capacity. Consequently, from (3.4), the final cost for edge $e_{n_{i_1},n_{i_2}}$ is

$$\mathcal{C}_{n_{i_1},n_{i_2}} = \frac{1}{c_{n_{i_1},n_{i_2}}}\frac{1-\prod_{k_j\in\mathcal{K}}\left[1-\exp\left(-\frac{P_\tau d_{n_{i_1},k_j}^{\alpha_{pl}}}{KP_{t,k_j}}\right)\right]}{\prod_{k_j\in\mathcal{K}}\left[1-\exp\left(-\frac{P_\tau d_{n_{i_1},k_j}^{\alpha_{pl}}}{KP_{t,k_j}}\right)\right]}. \tag{3.6}$$

3.2.4 Strength of the RPU Attack: A Toy Example

To demonstrate the strength of the proposed attack, a toy example is studied, which is illustrated in Figure 3.2. All of the nodes are located in a grid topology, and we assume that each link has a unit capacity no matter what the length of each link is. The black node is the primary user and white nodes are secondary cognitive radio devices. Source node n_S wants to transmit data to destination node n_D. The malicious node n_M can attract all the traffic from n_S and route the data towards the primary user (i.e., forwarding data to the nodes that are close to the primary users). There are two different routes from n_M to n_D, with the same number of hops, labeled as route 1

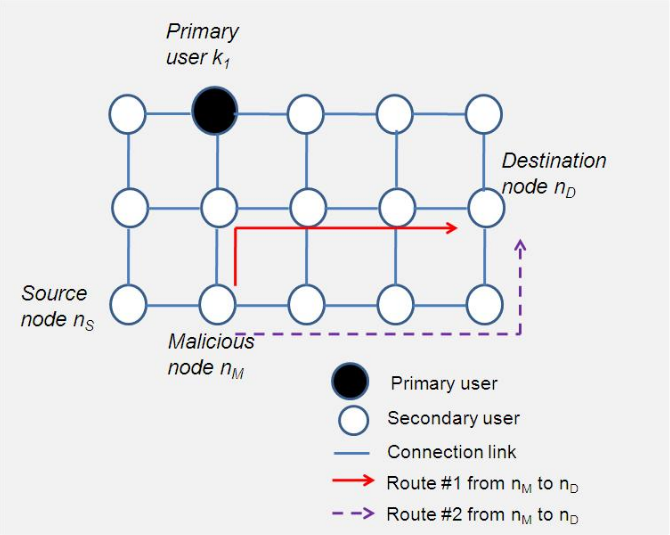

Figure 3.2 Toy example.

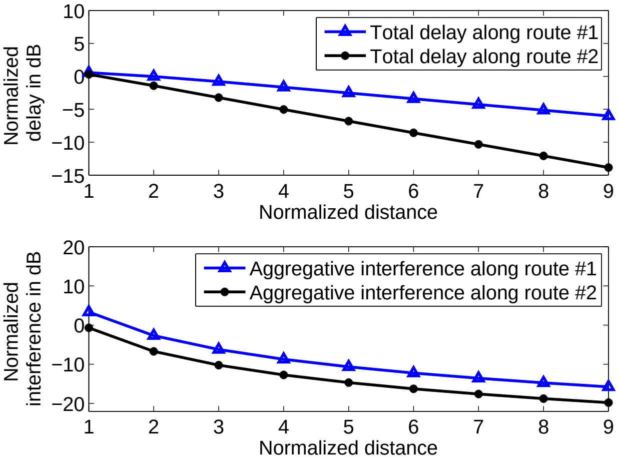

Figure 3.3 Effect of the RPU attack for the toy example.

and route 2, shown in Figure 3.2. Considering the position of the primary user, those two routes have different delays and different amount of interference to the primary user.

We simulated the data transmission using route 1 and route 2, respectively, and the simulation results are shown in Figure 3.3. We used the similar simulation setup as in Section 3.4. The x-axis is the length of each link, and the y-axis represents the total delay and the aggregative interference respectively. Even though both routes have the same number of hops, they have different delays and different aggregative interference. Because route 1 is closer to the primary user, the total delay and

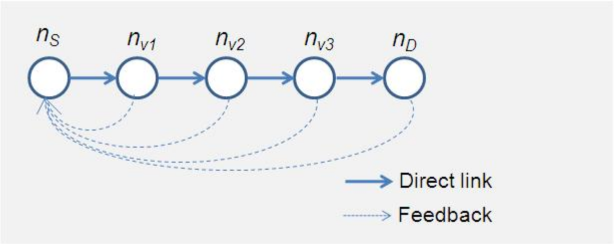

Figure 3.4 Illustration of the defense strategy against RPU attack.

interference is much higher than that of route 2. This example effectively demonstrates the damage of the RPU attack, in which attackers make the worse route (i.e., route 1) being chosen.

3.2.5 Routing Problem Formulation

Each source and destination pair will decide which route to choose. Because of some intermediate malicious nodes, which will try to forward the packets toward primary users and cause large delay in data transmission, the problem can be formulated as minimizing the delay through the route. For source-destination pair l, the utility function U_l can be calculated as

$$U_l = \sum_{e_{n_{i_1},n_{i_2}} \in R_l} \mathcal{C}_{n_{i_1},n_{i_2}}, \tag{3.7}$$

where R_l is one possible route for the source and destination pair l. The physical meaning of the utility function is the summation of delays along the route from the source to the destination. Therefore, the problem can be mathematically formulated as,

$$\min_{l} \; U_l. \tag{3.8}$$

Notice that the same philosophy can be used for other types of routing mechanisms in a similar way.

3.3 Defense Strategy Using Belief Propagation

In this section, we present the defense strategy for the RPU attack. Firstly, for indirect links in cognitive radio networks, the shortest path algorithm will be used to find the initial route from the source to the destination, without considering whether there are malicious nodes on the route or not.

Clearly, the initial route is a directed acyclic graph G with a set of nodes on the route indexed by $v \in V$. Using the concept of chain, for any node on the route, all its descendant nodes are called the nodes "after" it in the graph G. For example, in Figure 3.4, n_{v_2}, n_{v_3} and n_D are the nodes "after" node n_{v_1}, while the source node n_S is not "after" n_{v_1}. In order to find possible malicious nodes, each node on the route will send a certain amount of initialization packets to the nodes "after" it, and collect the feedback information from the nodes "after" it. The feedback information is also called m value, which is authenticated and encrypted, and cannot be modified by malicious nodes. Then each node will keep a table recording the feedbacks from all the nodes that are "after" it on the route. Clearly, the source node will keep the records of the feedbacks from all other nodes on the route.

In single-user decision making, only the local observation is used. To detect the malicious nodes, all neighboring nodes need to communicate with each other and exchange the feedbacks. A simple flooding strategy can be used, but the computation complexity and signaling overhead can be significant. To overcome this problem, we will use BP [71, 72, 84], which is an efficient way to calculate the marginal distribution and can avoid the involvement of other secondary nodes that are not on the initial route.

In the general framework of BP, the joint probability for the unknown variable X_v is defined as

$$\mathcal{P}(\{X_v\}) = \prod_{v=1}^{V} \phi_v(X_v) \prod_{v \neq w} \psi_{vw}(X_v), \tag{3.9}$$

where ψ_{vw} is called compatibility function and ϕ_v is called local function. In our case, we denote by X_v the state of node v. When $X_v = 1$, it is an honest node; otherwise, $X_v = 0$.

Our goal is to compute the marginal probabilities of each node, which can be mathematically defined as sums over all possible states of all its parents. For example, for node n_{v_1} in Figure 3.4, we can have

$$\mathcal{P}(n_{v_1}) = \sum_{n_{v_2}} \sum_{n_{v_3}} \sum_{n_D} \mathcal{P}(n_{v_2}, n_{v_3}, n_D). \tag{3.10}$$

We call this marginal probability "beliefs," and denote the belief at node v by b_v. A Bayesian network is a probabilistic graphical model that represents a set of random variables and their conditional dependencies. For small Bayesian networks, we can easily do marginalization sums directly. Nevertheless, when the number of nodes in the network increases, the number of terms in the sums will grow exponentially. This means that it is unreliable to do marginalization sums directly in large Bayesian networks. Therefore, we will use BP to compute marginal probabilities, which grows only linearly with the number of nodes.

BP can be used to compute marginal probability of node v iteratively. In the l-th iteration, node w sends a feedback $m_{wv}^l(X_v)$ to node v, which can be given by

$$m_{wv}^l(X_v) = \sum_{X_w} \psi_{wv}(X_w, X_v)\phi_w(X_w) \prod_{u \neq w,v} m_{uw}^{l-1}(X_w). \tag{3.11}$$

Intuitively, m_{wv}^l means the belief about the state of node w tested by node v.

In each iteration, each node computes its belief, based on the equation proposed in [59],

$$b_v(X_v) = k_v \phi_v(X_v) \prod_{w \neq v} m_{wv}^M(X_v), \tag{3.12}$$

where k_v is a normalization constant and $m_{wv}^M(X_v)$ denotes the feedback from node w to node v about what state node w should be in.

Theorem 3.1. *In the graph of chain, BP converges and gives the exact marginal probabilities.*

Proof. In the graph of chain, for any directed link $\mathbf{A} \to \mathbf{B}$, we can quantify it by a fixed conditional probability matrix M,

$$M_{\mathbf{b}|\mathbf{a}} = P(\mathbf{B} = \mathbf{b}|\mathbf{A} = \mathbf{a}) = \begin{pmatrix} P(\mathbf{b_1}|\mathbf{a_1}) & P(\mathbf{b_2}|\mathbf{a_1}) & \dots & P(\mathbf{b_n}|\mathbf{a_1}) \\ P(\mathbf{b_1}|\mathbf{a_2}) & P(\mathbf{b_2}|\mathbf{a_2}) & \dots & P(\mathbf{b_n}|\mathbf{a_2}) \\ \vdots & \vdots & & \vdots \\ P(\mathbf{b_1}|\mathbf{a_m}) & P(\mathbf{b_2}|\mathbf{a_m}) & \dots & P(\mathbf{b_n}|\mathbf{a_m}) \end{pmatrix}. \tag{3.13}$$

If we consider the simplest case for the chain, in which the network consists of only two nodes $\mathbf{A}$ and $\mathbf{B}$. We have only one directed link $\mathbf{A} \to \mathbf{B}$. If we define $\lambda(\mathbf{a})$ as the likelihood vector of $\mathbf{A}$ when evidence $\varepsilon = \{\mathbf{B} = \mathbf{b}\}$ is observed, the belief distribution of $\mathbf{A}$ can be given by

$$b(\mathbf{a}) = P(\mathbf{a}|\varepsilon) = \mu P(\mathbf{a})\lambda(\mathbf{a}), \tag{3.14}$$

where $\mu = [P(\varepsilon)]^{-1}$, and $P(\mathbf{a})$ is the prior probability of $\mathbf{A}$. $\lambda(\mathbf{a})$ can be given by

$$\lambda(\mathbf{a}) = P(\varepsilon|\mathbf{a}) = P(\mathbf{B} = \mathbf{b}|\mathbf{a}) = M_{\mathbf{b}|\mathbf{a}}. \tag{3.15}$$

Since this matrix is stored at node $\mathbf{B}$, $\lambda(\mathbf{a})$ can be computed at $\mathbf{B}$ and transmitted as a message to $\mathbf{A}$. Therefore, $\mathbf{A}$ can compute its belief distribution $b(\mathbf{a})$ easily.

Now, let's assume that there are three nodes $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ on the chain now, and $\mathbf{C}$ is the descendant of $\mathbf{B}$. In this case, we can have the chain $\mathbf{A} \to \mathbf{B} \to \mathbf{C}$, and $\mathbf{B}$ is not observed directly but is supported by indirect observation $\varepsilon = \{\mathbf{C} = \mathbf{c}\}$. We can still write

$$b(\mathbf{a}) = P(\mathbf{a}|\varepsilon) = \mu P(\mathbf{a})\lambda(\mathbf{a}). \tag{3.16}$$

Nevertheless, $\lambda(\mathbf{a})$ is no longer directly obtained from the matrix $M_{\mathbf{b}|\mathbf{a}}$, and must reflect the matrix $M_{\mathbf{c}|\mathbf{b}}$ as well. Therefore, we can write

$$\lambda(\mathbf{a}) = P(\varepsilon|\mathbf{a}) = \sum_{\mathbf{b}} P(\varepsilon|\mathbf{b}, \mathbf{a})P(\mathbf{b}|\mathbf{a}) = \sum_{\mathbf{b}} P(\varepsilon|\mathbf{b})P(\mathbf{b}|\mathbf{a}) = M_{\mathbf{b}|\mathbf{a}}\lambda(\mathbf{b}), \tag{3.17}$$

using the fact that $\mathbf{B}$ separates $\mathbf{A}$ from $\mathbf{C}$. Consequently, we have shown that node $\mathbf{A}$ can still calculate its likelihood vector $\lambda(\mathbf{a})$ if it somehow gains access to the vector $\lambda(\mathbf{b})$.

Therefore, in a general case, in which we have multiple nodes on the chain (two or more), if each node constantly inspects the λ of its child and updates its own λ accordingly, we are guaranteed that every variable along the chain will be able to calculate the correct value of its λ, properly reflecting any changes that might have occurred in ε. □

Based on Theorem 4.5, for the initial route in our case, which is a chain, we can have

$$b_v(X_v) = p(X_v). \tag{3.18}$$

If the final belief for a node is higher than a threshold, denoted by b_τ, this node is an honest node. In contrast, if the final belief for a node is lower than a threshold b_τ, this node can be seen as a malicious node. Therefore,

$$v = \begin{cases} \text{honest}, & b_v \geq b_\tau; \\ \text{malicious}, & b_v < b_\tau. \end{cases} \tag{3.19}$$

Note that each node only needs to send one packet to each of the other nodes "after" it. This packet is just used to initialize a feedback request from all the other nodes "after" it. The feedback packet is calculated using Eq. (3.11). In our specific problem, the feedback only contains an "m" value between 0 and 1. So the packet size is very small. Therefore, we can see that the overhead of communication is very low. Moreover, we want to emphasize that these transmissions only happen in order to catch the malicious node, and do not happen normally. In addition, one of the advantages of BP is that when there are many observations in each round of the detection, BP can automatically aggregate the observations into a much lower dimensional space of parameters to reduce the curse of dimension. In our proposed algorithm, only one parameter needs to be exchanged between the users in each message, while the direct observation exchange may incur substantial communication overhead. So the communication complexity of BP is only a linear function of the number of nodes on the route, i.e., $O(N)$.

3.3.1 Local Function

The feedback information can be defined as the probability that whether this node is an honest node or not. First, let us define the link quality between two nodes, which can be described by a *trust value* that represents how much a wireless link can be trusted to deliver packets correctly in the routing problem [85]. Trust value (link quality) of each link can be represented in the form of Beta function, which is often used in the scenarios where one node has collected binary observation about the other node. Assume that node n_{v_a} and node n_{v_b} are in each other's direct transmission range and node n_{v_a} sends a certain amount of packets to node n_{v_b}. Assume that $\alpha - 1$ packets from node n_{v_a} are received by node n_{v_b} successfully and $\beta - 1$ packets fail to arrive at node n_{v_b}, then the qualify of link $e_{n_{v_a},n_{v_b}}$ can be represented as

$$B(\alpha,\beta)_{e_{n_{v_a},n_{v_b}}} = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} p^{\alpha-1}(1-p)^{\beta-1}, \tag{3.20}$$

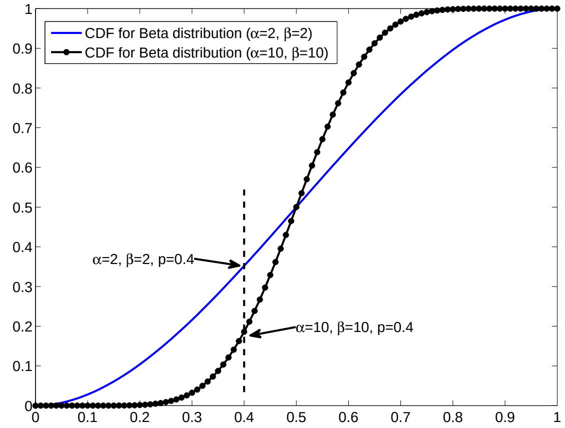

Figure 3.5 Comparison of the CDF curves for different α and β parameters.

where Γ is the gamma distribution, and a Bernoulli distribution with parameter p governs whether transmissions succeed or fail. In other words, the probability that n_{v_b} successfully forward packets is denoted by p. The mean and variance of the $B(\alpha, \beta)$ can be calculated as

$$\mu = \frac{\alpha}{\alpha + \beta}; \sigma = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}. \tag{3.21}$$

A probability density function of this type expresses the uncertainty that future interactions will succeed or fail. The cumulative distribution function (CDF) of Beta distribution is defined as

$$F(p; \alpha, \beta) \quad = \quad I_p(\alpha, \beta) = \sum_{j=\alpha}^{\alpha+\beta-1} \frac{(\alpha+\beta-1)!}{j!(\alpha+\beta-1-j)!} p^j (1-p)^{\alpha+\beta-1-j}, \tag{3.22}$$

where $I_p(\alpha, \beta)$ is called regularized incomplete Beta function. If we set a threshold p_τ for the Bernoulli parameter p in the CDF function, then the local function can be defined as

$$\phi_v = 1 - F(p \geq p_\tau; \alpha, \beta). \tag{3.23}$$

The physical meaning of the local function is the probability that the node who sends back the feedback is an honest node or not. Obviously, we have $0 \leq \phi_v \leq 1$, and the lower value of ϕ_v, the higher probability this node is a malicious node. Figure 3.5 shows the cumulative distribution functions (CDF) of Beta distribution for the two cases $\alpha = 2, \beta = 2$ and $\alpha = 10, \beta = 10$. Although in these two cases, they have the same mean value of $\frac{\alpha}{\alpha+\beta} = 0.5$, the variance in the second case is much smaller than that in the first case (since the number of observations is more). Therefore, the CDF curves are different. If we set an appropriate threshold for the Bernoulli parameter p, e.g. 0.4 in Figure 3.5, the probability of success transmission in the second case is higher than in the first cases, considering there are more trials in the second case. In addition, with the increasing of the values of α and β, as well as the ratio between α and β, the success rate increases, which means that the probability that this node is a malicious node will decrease, and the local function ϕ_v will increase.

3.3.2 Compatibility Function

It is prohibitively difficult to find an explicit expression for the compatibility function $\psi_{wv}(X_w, X_v)$, and we can only propose a heuristic one. We notice that $\psi_{wv}(X_w, X_v)$ is dependent on the correlation between states X_w and X_v. If X_w and X_v are highly positively correlated, i.e., they tend to be the same, $\psi_{wv}(X_w, X_v)$ should yield a large probability for $X_w = X_v$ and a small probability for $X_w \neq X_v$. Therefore, we propose to use the following simple compatibility function, which is given by

$$\psi_{wv}(X_w, X_v) = \left\{ \begin{array}{cl} \eta, & X_w = X_v, \\ 1-\eta, & X_w \neq X_v, \end{array} \right. \tag{3.24}$$

where $0.5 < \eta < 1$ is a predetermined value. The larger η is, the more correlation between neighboring nodes we assume. [2]

3.3.3 Complete Algorithm

The complete routing algorithm is summarized in Algorithm 3.1. In the first step, for a given source and destination pair, using the shortest path routing algorithm[3] with cost function defined in (3.6), a path can be found, which may include some malicious nodes. In the second step, each

[2] A reasonable value of η can be adjusted considering the possibility of bad-mouthing attacks, which is beyond the coverage of this chapter [86].

[3] We are just using the shortest path routing algorithm as an example, and other routing algorithms can be employed in a similar way.

Algorithm 3.1 Complete defense algorithm against RPU attack using belief propagation

```
1: Using the shortest path routing algorithm to obtain an initial route from the source to the destination.
2: Each node on the initial route keeps a table recording feedbacks from the nodes “after” it.
3: for Each iteration do
4:     Each node computes the local function using (3.23) and the compatibility function using (3.24).
5:     Each node computes $m$ values using (3.11).
6:     Each node exchanges $m$ values with neighbors.
7:     Each node computes its belief using (3.12).
8: end for
9: The source node detects the malicious nodes according to final beliefs.
10: Using the shortest path algorithm to find a new route avoiding those malicious nodes.
```

node on the initial route keeps a table used for recording feedbacks from other nodes on the route. Each feedback is just one calculated m value (see Equation 3.11). The memory for each node to keep the table is proportional to the size of the number of the nodes on the route, which makes the overhead very low. In addition, considering the fact that BP only grows linearly with the number of nodes, therefore, the computation complexity of our algorithm is also very low. From step 3 to step 7, in an iterative way, each node on the route will carry out the measurements and calculates the local functions, as well as the compatibility functions. Each node will then send some packets to the nodes “after” it, and collect the feedback information (i.e., m values) from all those nodes. They will also exchange feedbacks with the neighboring nodes and compute belief values until convergence. As a result, in step 9, the source node will detect the malicious node according to final beliefs. If the final belief values of some nodes are below a threshold, those nodes will be seen as malicious nodes. Finally, in step 10 the shortest path algorithm will be used to find a new route from the source to the destination, avoiding those malicious nodes.

3.4 Simulation Results and Analysis

In this section, we present the simulation results. We use Matlab to simulate the CR network, in which 1 primary user and 500 secondary cognitive devices are randomly deployed in a 100 meter-by-100 meter area. The maximum transmission range for each secondary user is 20 meters, and the path loss factor is set to be 2.5. The noise variance is set to be 0.01, the transmit power is set to be 1W, the parameter K is set to be 1, SNR threshold γ_{k_j} is set to be 10dB, Beta CDF threshold p_τ

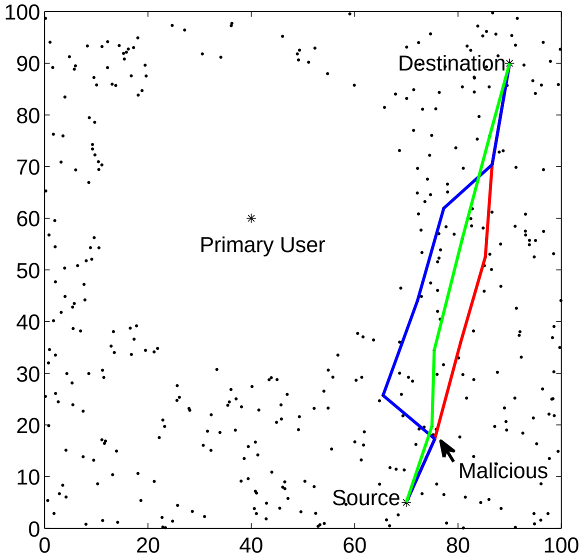

Figure 3.6 One snap shot of simulated topology from source to destination. One primary user and 500 secondary users are randomly deployed in a 100 meter-by-100 meter area.

is set to be 0.8, η in the compatibility function and the belief threshold b_τ are set to be 0.7 and 0.3, respectively.

Figure 3.6 shows the simulation results for one snap shot. The primary user is located at $(40m, 60m)$, and the source and the destination nodes are located at $(70m, 5m)$ and $(90m, 90m)$ respectively. We can see that in Figure 3.6, there is a blank region around the primary user, which represents the footprint of the primary user. The footprint depends on the fading channel model, and any secondary user that is inside this region is blocked to avoid interference to the primary user. Only the secondary users that are outside the footprint of the primary user are allowed to be turned on, shown as the black dots in Figure 3.6. There is only one malicious node in this case, which is located at $(75.52m, 17.30m)$. If all the nodes behave honestly, the source will find the route as shown in the red dashed path on the right. However, if the malicious node behaves dishonestly, it will route the packets to the node located at $(65.49m, 25.71m)$, which is close to the primary user, shown as the blue dashed route on the left. After the source node detects the malicious node using the proposed defense strategy, it will adjust the path, shown as the solid route, which avoids the malicious node.

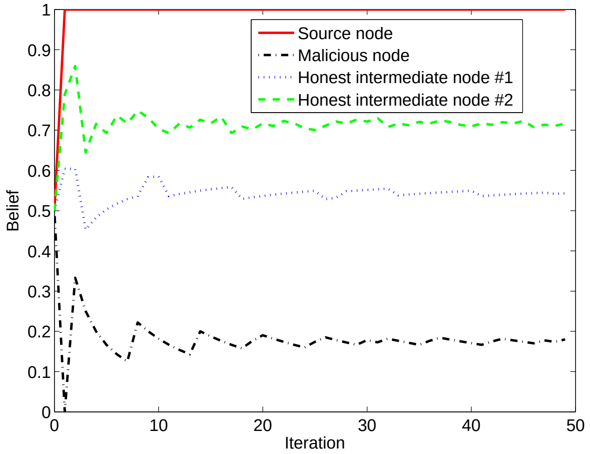

Figure 3.7 Simulation results about the beliefs over iterations for four different secondary nodes.

Figure 3.7 shows the simulation results for the case that there is only one malicious node on the route from the source to the destination. The x-axis represents iterations and the y-axis represents belief values. From the top down, the curves represent the belief of source node, belief of honest intermediate node #1, belief of honest intermediate node #2, and belief of the malicious node, respectively. We can see that the convergence for belief propagation is within a couple of loops. Also there is oscillation in the process of convergence because of the existence of inaccurate messages. We can also find that for the source node, the belief value will converge to 1, and for the other two honest intermediate nodes, their belief values converge to a value above 0.5. However, for the malicious node, the final belief value will be at around 0.2. In this way, we can set a belief threshold of 0.3, and the malicious node can be detected easily. In addition, for the case if there are more than one malicious node on the initial route, based on simulation results, only the first malicious node "after" the source on the route can be detected, while the final belief values of other malicious node "after" the first malicious node may be above the threshold. To avoid the first malicious node, the source will re-route and get a new path to the destination. There is a high probability that those malicious nodes "after" the first malicious node on the initial route will not be on the new route. If some malicious nodes are still on the new route, the source node needs to perform detection again. Therefore, to make sure that we can avoid all malicious nodes, we need to run a certain number of iterations. As far as the final beliefs of all the nodes on the route are above

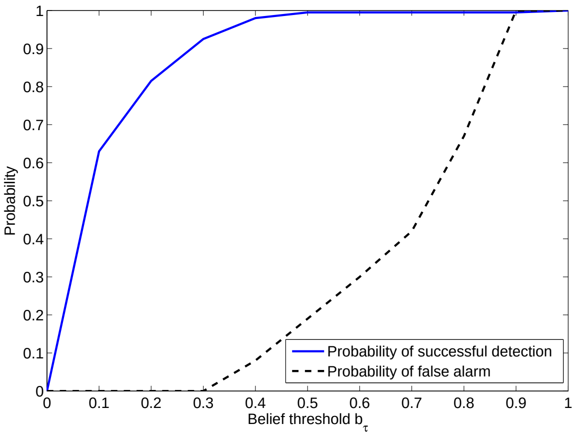

Figure 3.8 Simulation results for the probability of successful detection and the probability of false alarm given different belief thresholds.

the threshold, we can stop the iteration and know that all the nodes on the route are honest nodes. This does not mean that the system performance or the routing speed is degraded. After finding the malicious nodes, all nodes can keep a table recording the information of malicious nodes and exchange the information with honest neighboring nodes. Therefore, in the future when some nodes want to find a path to other nodes, they will automatically avoid those malicious nodes that have already been detected based on their records.

Figure 3.8 shows the probability of successful detection of malicious nodes and the probability of false alarm when the belief threshold changes. We can find that as the belief threshold b_τ increases, the probability of successful detection increases. When b_τ is 0.3, successful detection rate is 0.91 and the probability of false alarm is 0. When b_τ is above 0.5, we can have 100% detection rate. However, when b_τ is bigger than 0.3, the probability of false alarm will not be 0 anymore. Therefore, when we decide the value of b_τ, we need to consider the tradeoff between the successful detection rate and the probability of false alarm.

Figure 3.9 shows the simulation results for delay when the number of malicious nodes increases. The x-axis is the number of malicious nodes and the y-axis represents the delay which is normalized based on the delay of 15 meters. From the top down, the curves respectively represent the delay with the malicious nodes having attack probability 1, with the malicious nodes having

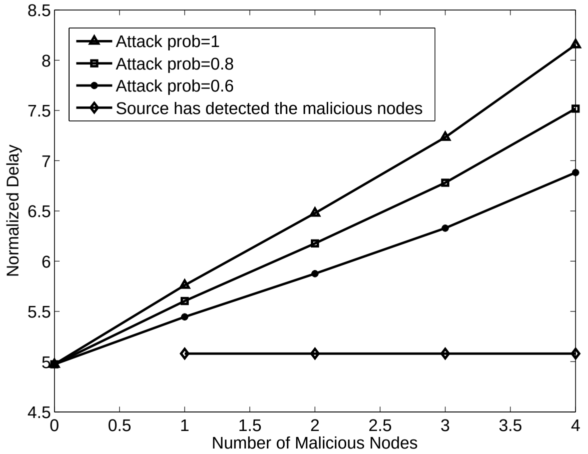

Figure 3.9 Simulation results for the delay along the route when the number of the malicious nodes changes.

attack probability 0.8, with the malicious nodes having attack probability 0.6 and after the source node detects the malicious nodes and re-routes avoiding those malicious nodes. Here, attack probability is the probability that the malicious node will conduct the RPU attack. It is clear that delay will increase with the increasing of the number of malicious nodes, as well as when the attack probability increases. In addition, from the curve representing the delay after source node detects the malicious nodes and re-routes, we can find that the delay is greatly decreased, which can be 3.1dB lower than if we have 4 malicious nodes. However, the delay of the new route is slightly higher than that in the case there are no malicious nodes, about 0.1dB. This is because when the source node tries to avoid the malicious nodes, it may re-route to some nodes that are slightly closer to the primary user, or the number of intermediate nodes along the route may increase. In addition, when there is no malicious node in the network, the source node does not need to re-route.

In Figure 3.10, the system performance in terms of delay is shown when the probability of attack increases. From the top down, the curves represent the cases of four malicious nodes, three malicious nodes, two malicious nodes, one malicious node and no malicious node, respectively. We can find that delay increases when the probability of attack increases, which makes perfect sense. Compared to the situation that there is no malicious node, the delay can be 4.7dB higher in the case if we have four malicious nodes. Note that this is the simulation result for only one source-

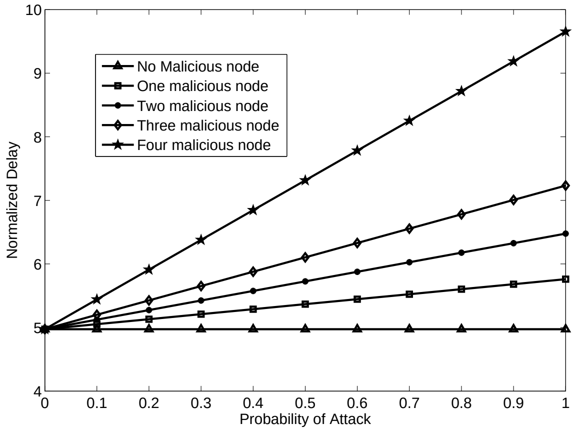

Figure 3.10 Simulation results for the delay along the route when the probability of attack changes.

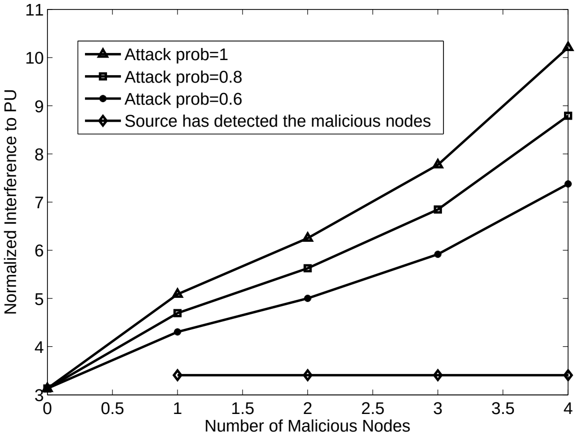

Figure 3.11 Simulation results for the aggregate interference to the primary user when the number of the malicious nodes changes.

destination pair, and if there are multiple source-destination pairs, the difference will be even bigger. In addition, when the number of malicious nodes increases, the delay also increases.

Figure 3.11 and Figure 3.12 show the simulation results about the aggregate interference to the primary user when the number of malicious nodes and attack probability changes, respectively. Aggregate interference is calculated by adding the interference from all the secondary users along the route. Also in the results, the interference is normalized based on the interference from 1 secondary user localized at a distance of 50 meters away from the primary user. Clearly, we can find that when the number of malicious nodes increases or the attack probability increases, the aggregate

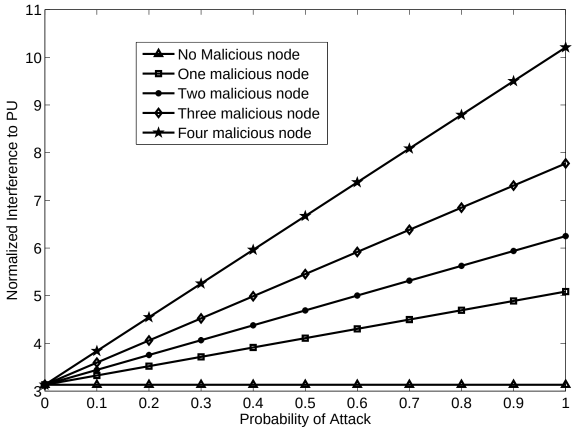

Figure 3.12 Simulation results for the aggregate interference to the primary user when the probability of attack changes.

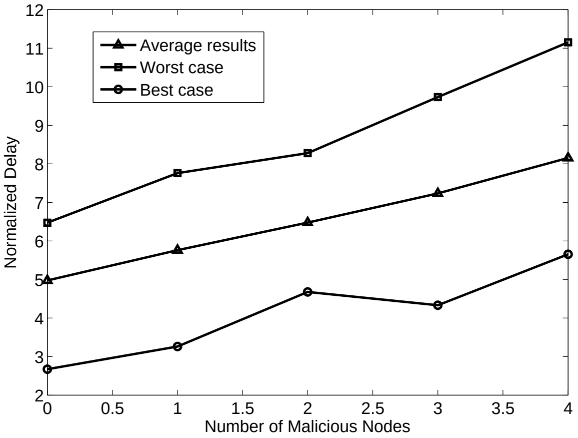

Figure 3.13 Different from Figure 3.9, the average results and the results of the worst case and the best case about delay along the route are shown when the number of the malicious nodes changes given attack probability of 1.

interference to the primary user increases. The aggregate interference can be reduced by 7dB for only one source-destination pair, which justifies our claim about the harm of the RPU attack and the effectiveness of the proposed defense scheme. These results are consistent with the results shown in Figure 3.9 and Figure 3.10.

In Figure 3.13 and Figure 3.14, the average results and the results of the worst case and the best case of 1000 simulations about delay along the route as well as aggregate interference are shown, when the number of malicious nodes changes. Best case means that the malicious nodes are close to the boundary of the primary user's footprint, and worst case represents that the malicious

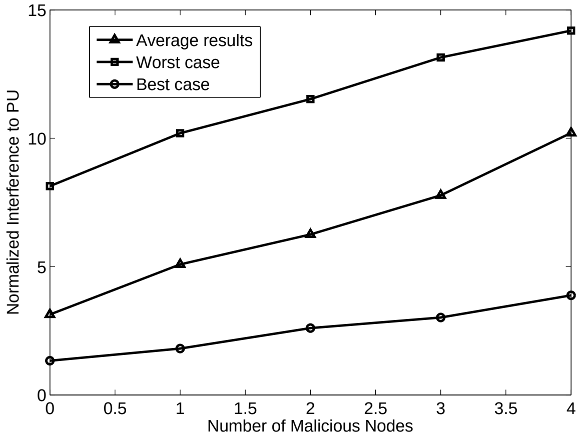

Figure 3.14 Different from Figure 3.11, the average results and the results of the worst case and the best case about aggregate interference are shown when the number of the malicious nodes changes given attack probability of 1.

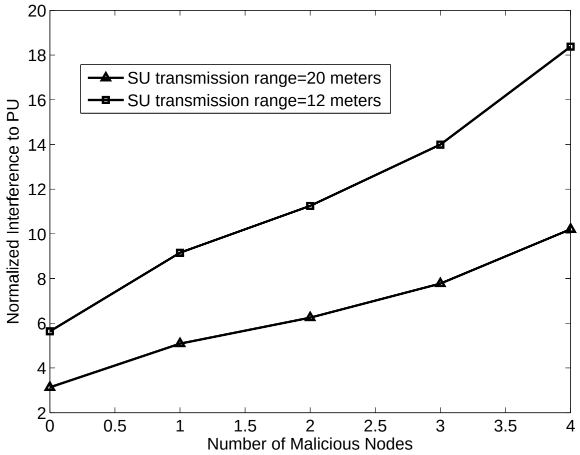

Figure 3.15 Simulation results about the aggregate interference along the route when the number of the malicious nodes changes given different secondary transmission ranges.

nodes are far away from the primary user. From the simulation results, we can see that different locations of the malicious nodes may have great impact on the performance of attack. Given a certain source and destination pair, when the malicious nodes are close to the boundary of the primary user's footprint, the performance degradation is not so big compared to the case that the malicious nodes are far away from the primary user.

Figure 3.15 shows the simulation results of the aggregate interference along the route, when the number of the malicious nodes changes for two different secondary transmission ranges. We can find that when the secondary transmission range decreases, the aggregate interference increases. This is because when the secondary transmission range decreases, the number of nodes along the route increases.

3.5 Conclusion

In this chapter, we proposed a new network layer attack, routing-toward-primary-user attack, in cognitive radio wireless networks. In the RPU attack, malicious nodes intentionally route a large amount of packets toward the primary users, aiming to cause large interference to the primary users and to increase delay along the routes. In the RPU attack, the malicious nodes “neutralize the route by the hands of the others,” and it is hard to detect the malicious nodes. To overcome this attack, we developed a defense strategy using belief propagation. Firstly, an initial route will be found from the source to the destination without the knowledge of the malicious nodes. Each node on the route keeps a table recording the feedback from all the other nodes “after” it on the route. Then in each iteration, every node exchanges the m values with its neighboring nodes and computes beliefs. After convergence, the source node can detect the malicious nodes based on the final belief values. Finally, a new route will be found, avoiding malicious nodes. Simulation results show that the convergence of BP is very fast, and the proposed defense strategy against the RPU attack is effective and efficient. Delay can be significantly reduced by 4.7dB for only one source-destination pair and the aggregate interference to the PUs can be reduced by 7dB caused by the RPU attack.

Chapter 4

Interference Aware Routing Using Network Formation Game in Cognitive Radio Mesh Networks

In this chapter, we propose the interference aware routing algorithm using the network formation game in cognitive radio mesh networks. The proposed distributed algorithm minimizes the aggregate interference from the secondary users to the primary users, and can avoid the problems in the centralized routing solution, such as the high cost for building the centralized coordinate nodes, high information-gathering delay, and system breakdown caused by the possible failures in the centralized nodes. Simulation results show that the proposed scheme finds better routes in terms of interference to the primary users compared to the shortest path scheme, and the distributed solution shows near optimum compared to the centralized solution. In this chapter, Section 4.2 introduces the cognitive radio wireless mesh network model. In Section 4.3, we provide the formulation of the distributed routing algorithm. In Section 4.4, the simulation results are presented, and finally the conclusion is drawn in Section 4.5.

4.1 Introduction

In cognitive radio (CR) wireless mesh networks (WMNs), wireless mesh routers work as secondary users (SUs) that can opportunistically and intelligently access the idle licensed spectrum. Though the interference from a single SU that is outside the PUs' footprints is small, the aggregate interference from a great number of SUs transmitting at the same time may be significant, and this will greatly influence the PUs' performance. When developing efficient routing techniques in CR networks, the aggregate interference from the SUs to the PUs should be considered.

We have introduced the existing routing algorithm for CR networks in Chapter 1. However, few in the literatures consider the aggregate interference to the PUs from a large amount of SUs, especially when the SUs are transmitting at the same time. Also the game theoretical approaches have been less investigated in the routing problems for the CR networks. In this chapter, we develop routing algorithms for CR-WMNs to minimize the cumulative interference from the SUs to the PUs. Note that we are not considering the interference between different SUs or between multiple paths, which has been well investigated in the idea of interference aware routing [87]. Instead, we are studying the aggregate interference from multiple SUs to the PUs in the CR networks. In CR-WMNs, PU's footprint is defined as an area that no SU is allowed to be turned on in order to avoid the interference from the nearby SUs to the PUs. When the SUs are outside the PUs' footprint, they can utilize the cognitive functionalities to access the licensed spectrum. However, even if a single SU is out of the PUs' footprint, and the interference from this single SU is not high, the cumulative interference from a lot of SUs transmitting at the same time to the PUs can still be significant, and the performance of the PUs can be greatly influenced by this aggregate interference. We develop a distributed algorithm using the network formation game framework and a myopic distributed algorithm to minimize the aggregate interference from the SUs to the PUs [88]. We also formulate a centralized optimization using linear programming, which serves as an upper bound for our distributed algorithm. While the centralized solution may produce a high cost for building the centralized coordinate nodes or cause the system breakdown because of the possible failures in the centralized nodes, our distributed algorithm is more practically implementable and reliable. From the simulation results, we can see that the proposed distributed algorithm's performance is very close to the optimum compared with the upper bound. Also, compared to Dijkstra's routing algorithm, it can find better routes with lower interference to the PUs. In summary, the main contributions can be outlined as follows:

- Different from the literatures, we consider the aggregate interference from a large amount of SUs to the PUs for routing problems in CR networks. At the same time, our routing algorithms are also developed to avoid high transmission delay along the routes.
- We use a game theoretical model to develop the distributed algorithm using network formation game. Our proposed distributed algorithm is practically implementable.

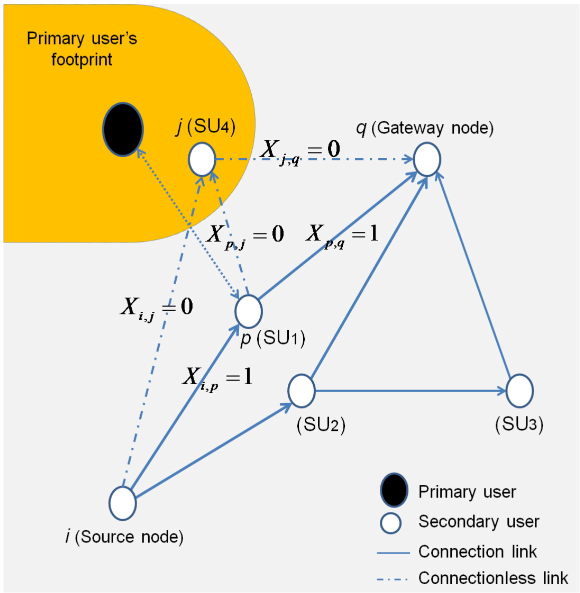

Figure 4.1 System model for CR-WMNs.

- We also develop a centralized algorithm for CR-WMNs, which serves as an upper bound. Simulation results verifies that the performance of the proposed distributed approach is very close to optimum.

4.2 Cognitive Radio Wireless Mesh Network Model

In CR-WMNs, the wireless routers work as the SUs which are equipped with CR functionalities. They have the capabilities to sense the spectrum and access the idle spectrum holes left by the PUs, as far as they can guarantee that their transmissions will not interfere with the PUs' signal. The SUs can employ the spectrum sensing techniques, such as radio identification based sensing or the spectral correlation algorithm, to detect the unoccupied licensed spectrum [89]. $\mathcal{N}$ is defined as the set of SUs in the CR-WMNs, and any SU $i \in \mathcal{N}$. We also define $\mathcal{E}$ as the set of the direct links. Flow on direct link $e \in \mathcal{E}$ is defined as f_e. If two SUs are in each other's transmission range, we define the link between these two nodes as a direct link. Otherwise, the link is called an indirect link, and intermediate nodes along the link are required to relay packets. The capacity of direct link

$e = (i, j)$ is defined as $c_{i,j}$, and it can be calculated using

$$c_{i,j} = W \log_2 \left(1 + \frac{P_i d_{i,j}^{-\alpha} h}{N_j + \Gamma} \right), \tag{4.1}$$

where $d_{i,j}$ is the distance between node i and j, W represents the bandwidth, α is path loss constant, P_i is the transmission power of node i, and h is the channel response that can be defined as a circular symmetric complex Gaussian random variable. N_j and Γ represent AWGN noise and the interference from other nodes, respectively. We also define an indicator variable $X_{i,j}$, which indicates the connectivity of link $e = (i, j)$. When $e = (i, j)$ is active, $X_{i,j} = 1$; Otherwise, $X_{i,j} = 0$. The CR-WMN model is illustrated in Figure 4.1, and we can see that the big solid node is a PU with the shadowed region representing the PU's footprint. White nodes are SUs, and solid lines represent the links that are connected and dashed lines are the links that have no connections. If the PU is occupying the licensed spectrum, secondary users, such as SU4, which are inside the PUs' footprint, are not allowed to access the spectrum. Therefore, we can have $X_{i,j} = 0$, $X_{p,j} = 0$, and $X_{j,q} = 0$. In contrast, if SUs are out of PU's footprint, SUs such as nodes i, p, and q are allowed to access the spectrum. Consequently, we can have $X_{i,p} = 1$ and $X_{p,q} = 1$, showing that the SUs can access the spectrum because they are out of the PU's footprint and the interference from single SU to the PU is sufficiently low.

4.2.1 Routing with Minimum Aggregate Interference to the PUs in CR-WMNs

When the distance between a single SU and the PUs is sufficiently long, it may produce sufficiently low interference to the PUs . Nevertheless, when the number of the SUs increases, and a large amount of SUs are transmitting at the same time, the aggregate interference from the SUs to the PUs can be significant. When designing routing protocols in CR-WMNs, we need to consider to minimize this aggregate interference. The concept of interference temperature can be considered to model the interference level in CR-WMNs [90]. We use the generalized interference temperature model T_I, i.e,

$$T_I(f_c, B) = \frac{\mathcal{P}_I(f_c, B)}{kB}. \tag{4.2}$$

In the above interference temperature equation, B is the bandwidth in Hertz, and $\mathcal{P}_I(f_c, B)$ represents the average interference power in Watts centered at frequency f_c. k represents Boltzmann's constant, and its value is 1.38×10^{-23} Joules per Kelvin degree.

Let's have a look at Figure 4.1 again. We can find that because SU1 is located closer than SU2 to the PU, the interference temperature level of SU2 is lower than that of SU1. When SU i and SU q want to communicate with each other, they can transmit either through SU2 or through SU1. On the route from SU i to SU q, the aggregate interference includes interference level from node i, node q and the other SUs along the path between them. In the proposed algorithm, SU i should choose the path of $i \rightarrow$ SU2 $\rightarrow q$ instead of the shortest path of $i \rightarrow$ SU1 $\rightarrow q$ to minimize the interference to the PU.

4.2.2 Interference Range and Transmission Range

We define the channel gain between two secondary nodes i and j as $G_{i,j} = \beta d_{i,j}^{-\alpha}$, where d_{ij} is the distance between SU i and SU j, α is the path loss constant, and β is a constant related to antenna design. Only when the received power is higher than a threshold ρ_T, the data can be seen as successfully transmitted. We also assume that interference from a single SU is sufficiently small when received power at the PUs is lower than another threshold ρ_I. Therefore, the transmission range for a SU i can be calculated as $R_{T_i} = (\beta P_i / \rho_T)^{1/\alpha}$, where P_i is the transmission power of SU i. The interference range for SU i can be calculated in a similar way as $R_{I_i} = (\beta P_i / \rho_I)^{1/\alpha}$. When a SU is at least R_{I_i} away from the PUs, the interference from this single SU to the PUs can be seen as small enough, and it is allowed to be turned on and to sense and access the licensed spectrum. However, consider the case that a large number of SUs are transmitting at the same time, and even though all the SUs are out of the primary footprint, the aggregate interference from these SUs to the PUs can still be significant. Therefore, when designing the routing algorithms for CR-WMNs, we need to consider the aggregate interference from a large number of SUs to the PUs.

4.3 Distributed Routing Algorithm Using Network Formation Game

In this section, we propose a distributed algorithm using the network formation game for CR-WMNs. Compared to the centralized routing solution, which may cause problems such as the high cost for building the centralized coordinate nodes, high information-gathering delay, and system breakdown caused by the possible failures in the centralized nodes, the network formation based distributed routing algorithm can significantly reduce the system overhead and the computation complexity.

4.3.1 Game Formulation

Network formation game from the game theory provides a perfect framework to model the interactions among the SUs in CR-WMNs when they are trying to find the routes [88]. In the network formation game, players interact with each other in order to form a suitable graph that connects them together. A network graph G is formed based the interactions between the players and their decisions depending on the objectives and incentives of the players. Different from other types of games, the links or the connections between the players play very important roles in the network formation game. Therefore, we can model the routing problem in CR-WMNs as a network formation game, and SUs are players in the game. A directed graph $G(\mathbb{N}, \mathbb{E})$ will be formed as the result of the players playing the game. We define $\mathbb{N} = \{1, ..., N\}$ as the set of all SUs, and $\mathbb{E}$ denotes the set of edges between the SUs. Then we have the following definition,

Definition 4.1. *A path between two SUs i and j in the graph G can be defined as a sequence of SUs i_1,...,i_K. Then we will have $i_1 = i$, $i_K = j$, and for each $k \in \{1, \ldots, K-1\}$, the directed link $(i_k, i_{k+1}) \in G$. $\mathbb{V}_i$ is defined as the set of all paths from SU i to the destination of SU i ($\mathcal{D}_i$). Thus, $|\mathbb{V}_i|$ represents the number of paths from SU i to its destination $\mathcal{D}_i$.*

Convention Each destination $\mathcal{D}_i$ is connected to its source i through at least one path. Therefore, $\forall i \in \mathcal{N}$, we have $|\mathbb{V}_i| \geq 1$.

We need to define the strategy and the strategy space for each player in the network formation game. The strategy of SU i is to select the link that it wants to form from its strategy space, which can be defined as the SUs in $\mathbb{N}$ that SU i is able to and wants to connect to. We set a rule that player i is not allowed to connect to another player j, which is already connected to i. Mathematically, this means that if a link $(j,i) \in G$, then link (i,j) cannot be in the graph G. We define $\mathbb{A}_i = \{j \in \mathbb{N} \backslash \{i\} | (j,i) \in G\}$ as the set of nodes from which node i accepted a link (j,i), and $\mathbb{S}_i = \{(i,j) | j \in \mathbb{N} \backslash (\{i\} \bigcup \mathbb{A}_i)\}$ as the set of links corresponding to the nodes with whom node i wants to connect. Consequently, the strategy of player i is to select the link $s_i \in \mathbb{S}_i$ that it wants to form by choosing the player that it wants to connect to.

4.3.2 Utility

The players try to make decisions in order to maximize their utilities. Given a selected strategy s_i for any player $i \in \mathcal{N}$ and a network G, the utility of player i can be expressed as

$$u_i(G) = -B_e^1 B_e^2 T_{I_i} \frac{f_{i,i_{nexthop}}}{c_{i,i_{nexthop}}}, \tag{4.3}$$

where B_e^1 and B_e^2 are the barrier functions, T_{I_i} is node i's interference temperature, $f_{i,i_{nexthop}}$ is the flow on the edge between node i and its next hop, and $c_{i,i_{nexthop}}$ represents the capacity of the same edge.

We know that the flow on each edge cannot be bigger than the link capacity, which means that

$$f_e \leq c_e, \forall e \in \mathcal{E}. \tag{4.4}$$

In addition, the outgoing flow should be equal to the sum of incoming flow and generated traffic. Therefore, we can have

$$l_j + \sum_{e=(i,j)\in\mathcal{E}} f_e = \sum_{e=(j,i)\in\mathcal{E}} f_e, \tag{4.5}$$

where l_j represents the generated traffic of secondary node j. This is the flow conservation constraint. We assume that that l_j consists of only generated traffic if there is no incoming traffic from the wired Internet. Therefore, the barrier functions that consider the constraints can be defined as

$$B_e^1 = \left(\sum \frac{1}{1 - \frac{f_e}{c_e} + \varepsilon_1} \right)^{\kappa_1}, \tag{4.6}$$

and

$$B_e^2 = \left(\sum \frac{1}{1 - \frac{l_j + \sum_{e \in E} f_e}{\sum_e f_e} + \varepsilon_2} \right)^{\kappa_2}, \tag{4.7}$$

where κ_1 and κ_2 are used to weight the two constraint functions and their values are set to be greater than 0. ε_1 and ε_2 are two small dummy constants so that the denominators in the barrier functions are not zero. When the constraints are almost not met, the value of the constraint function will be large. Therefore, in the proposed utility function, the barrier functions protect the interference to the PUs to ensure that the two constraints in Eq. (4.4) and Eq. (4.5) are satisfied.

4.3.3 Proposed Algorithm for Network Formation Game

Now we will start to design an algorithm of interaction to form the network graph considering the utility function. When SU i plays a strategy $s_i \in \mathbb{S}_i$ and all other SUs keep their strategies $s_{-i} = [s_1, ..., s_{i-1}, s_{i+1}, ..., s_M]$, we can form graph $G_{s_i, s_{-i}}$. For each player i, although it can select strategy $s_i = (i, j) \in \mathbb{S}_i$ to maximize its utility, the other player j may refuse to accept this formation of s_i if this leads to a great decrease in its utility u_j. If the formed graph is defined as $G + s_i$ after player i selects strategy s_i and adds the link (i, j), we can define the concept of a feasible strategy as follows,

Definition 4.2. *A strategy $s_i \in \mathbb{S}_i$, i.e., a link $s_i = (i, j)$ is a feasible strategy for player i if and only if $u_j(G_{s_i, s_{-i}} + s_i) \geq u_j(G_{s_i, s_{-i}})$. For player $i \in \mathbb{N}$, we can define the set of all feasible strategies as $\breve{\mathbb{S}}_i \subseteq \mathbb{S}_i$.*

A feasible strategy for player i represents a link between i and j that player j would like to accept the connection from i. Therefore, a feasible strategy for player i in the network is to form a link with another player j that will accept a connection with player i, as well as to maximize its utility u_i. On the other hand, any player j would like to accept a connection invitation from any other player i as far as the utility of player j will not decrease.

For any player $i \in \mathbb{N}$, given the feasible strategies set $\breve{\mathbb{S}}_i$, we can use the following definition to define the best response for a player,

Definition 4.3. *For a player $i \in \mathbb{N}$, a strategy $s_i^* \in \breve{\mathbb{S}}_i$ is defined as a best response if $\forall s_i \in \breve{\mathbb{S}}_i$, $u_i(G_{s_i^*,s_{-i}}) \geq u_i(G_{s_i,s_{-i}})$. Therefore, when the other nodes maintain their feasible strategies, the best response for player i is to choose the feasible strategy that can maximize its utility. The best response is to make the section of the link and it is a link replacement action.*

Algorithm 4.1 Proposed distributed algorithm using network formation game

```
while The final network has not converged and some players can still improve their utilities by
unilateral change of strategies do
    Phase 1 - Prioritization Phase:
        Authorize the priorities to the nodes based on their interference to the PUs.
    Phase 2 - Myopic Network Formation Phase:
        By the order in Phase 1, the wireless mesh SUs start the network formation process.
            a) Each SU $i$ engages in pairwise negotiations with the other nodes in $\mathbb{N} \setminus (\mathbb{A}_i \bigcup \{i\})$
               to access its utility in its turn.
            b) After negotiations, each player $i$ plays its best response $s_i^*$ from the feasible
                strategy space, maximizing its utility.
end while
```

Subsequently, a distributed algorithm for the formation of the network graph is proposed, as shown in Algorithm 4.1. We assume that the network is dense enough. Also we assume each player does not consider the future evolution of the network when it wants to maximize its utility or payoff. Each player only considers the current state of the network, and the players are called myopic players. The proposed myopic network formation algorithm consists of two phases: a fair prioritization phase and a network formation phase. In the fair prioritization phase, a priority function is developed and it assigns a priority to each node. In the network formation phase, the players interact to select the next hop to this destination by increasing priority.

In the proposed algorithm for the network formation, the first step of each round is the fair prioritization phase. In this phase, based on the player's interference to the PUs, we assign each player with a priority: the node with a higher interference to the PUs is assigned with a higher priority. The objective of the prioritization is to make the SUs that are close to the PUs and produce high interference to the PUs have an advantage when they select the path towards their destinations. Therefore, those players can have a larger chance to improve their performances because they are allowed to select their partners from a larger space of strategies. In addition, we need to emphasize that we can also utilize other priority functions. In fact, in the simulation results, we use a random priority function for a general case.

In the myopic network formation phase, the secondary nodes start to select their strategies from the strategy space based on the priorities defined in the fair prioritization phase. Given the formed network graph resulting from the strategies of all other players, player i plays its best response $s_i^* \in \breve{\mathbb{S}}_i$ in order to maximize its utility at each round. Every node replaces its current link to the destination with another link that maximizes its payoff, and therefore, the best response action is a link replace operation. In order to find the best response, each node engages in pairwise negotiations with the other nodes. After the negotiations , the node will select the strategy from the feasible strategy space that maximizes its utility. The algorithm will run for several rounds until convergence, and finally a graph $G^\sharp$ will be formed in which no player can improve its utility by changing the best response.

Definition 4.4. *A network graph G in which no player i can improve its utility by a unilateral change in its strategy $s_i \in \breve{\mathbb{S}}_i$ is a Nash network.*

From the definition above, we can see that when the link chosen by each node is the best response, a Nash network is formed. In a Nash network, no node is able to improve its utility by unilaterally changing its current strategy, which means that all the players are in a Nash equilibrium. Consequently, we can have $u_i(G_{s_i^*,s_{-i}}) \geq u_i(G_{s_i,s_{-i}}), \forall s_i \in \breve{\mathbb{S}}_i$, for any $i \in \mathcal{N}$.

In the following theorem, we prove that the investigated network is a Nash network.

Theorem 4.5. *In the game with finitely many nodes, there exists a Nash network $G^\sharp$.*

Proof. We denote a network formation game Ξ, which defined by $\Xi = \{\Xi_{h\in\mathbb{E}}\}$. E_{ij} and $\mathbb{E}$ denote the link from node i to node j and the set of links, respectively. Suppose the network maintains the same network graph, then Ξ's row indicates a spatial network formation game Ξ_h.

To find a multi-hop connection to destination node $\mathcal{D}_i$, node i needs to find a node to connect to at the next hop i_h. The chosen node (i, i_h) can potentially yield an optimal path with maximum utility leading to the local stage utility to node i is $u_i[(i, i_h), (-i, i_h)] : \mathbb{A}_i \times \mathbb{S}_i \to \mathcal{R}$. It is an instantaneous utility at the local stage i, which depends on the local actions of all nodes at i. The coupling among utility functions u_i induces a noncooperative environment, in which node i computes with other node $-i$ to achieve optimal utilities. Node i needs to choose an optimal path that gains the optimal utility. However, it can only choose from $\mathbb{A}_i$. The path utility at node i is given by

$$U_i[i, E(h)] = \sum_{i=1}^{E(h)} u_i(G)[(i, i_h), (-i, i_h)], \tag{4.8}$$

where $E(h) \subseteq \mathbb{E}$. Let $U_i^*[i, E(h)]$ be the optimal utility. The path utility (4.8) can be rewritten as

$$\begin{aligned} U_i^*[i, E(h)] = & \quad \max \sum_{i=1}^{E(h)} u_i(G)[(i, i_h), (-i, i_h)] \\ = & \qquad \max u_i(i, i_h) + U_i^*[i_h, E(h)]. \end{aligned} \tag{4.9}$$

In the right side of (4.9), U_i^* can be seen as the payoff-to-go and u_i is the current instantaneous payoff to optimize.

Lemma 4.6. *Backward induction requires complete knowledge of the topology. In a game with finitely many nodes, backward induction always results in a Nash equilibrium [91].*

Based on Lemma 4.6, a solution to (4.9) can be found using backward induction, where we start with nodes at the last connection node and then propagate the solution to node i. Since every node optimizes its utility in the same way, the best response of (4.8) to other nodes can lead to a Nash equilibrium, in which no one wishes to deviate from their chosen actions. □

After solving the network formation algorithm and obtaining the whole network topology, the source node may have several choices to the destination, as defined in Convention 1. However, if we select a route that is very far away from the primary users, which may provide significantly low interference to the primary users, we may have a large delay along this route. Therefore, we need a tradeoff between the cumulative delay and the aggregate interference. In order to make sure that the interference to the PUs is low enough without increasing much delay, we will select a route based on the constraint,

$$D_{total} \le D_{\tau}, \tag{4.10}$$

where D_{total} represents the total delay along the route, and D_{τ} is the threshold. Note that for different source and destination pairs, we may have different values for the delay threshold. Given the constraint in Eq. (4.10), the source will then select the route with the lowest aggregate interference to the PUs.

4.3.4 Centralized Algorithm

In this subsection, we formulate a centralized routing algorithm for CR-WMNs. We want to provide an upper bound for the distributed algorithm using network formation game, and we can compare the performance of the proposed algorithm with the centralized algorithm to tell how well the proposed algorithm works.

In the centralized routing algorithm for CR-WMNs, we want to to minimize the aggregate interference from the SUs to the PUs. The objective function can be defined as the product of the interference to the PUs and system resources,

$$I_e = \frac{f_e}{c_e} \times T_{I(e)}. \tag{4.11}$$

Therefore, we can formulate the optimization problem as

$$\min_{f_e} \sum_{e \in \mathcal{E}} I_e \tag{4.12}$$

$$\text{s.t.} \begin{cases} f_e \le c_e, \forall e \in \mathcal{E}, \\ f_e = 0, \forall e(i,j), \text{if } \; X_{i,j} = 0, \\ l_j + \sum_{e=(i,j)\in\mathcal{E}} f_e = \sum_{e=(j,i)\in\mathcal{E}} f_e. \end{cases}$$

There are three constraints in the centralized optimization, and we can find that the constraint 1 and constraint 3 are from Eq. (4.4) and Eq. (4.5). The second constraint is saying that if SU i and

SU j want to communicate with each other, there must be a common licensed channel available for both of them. In (4.12), the variable is flow, and the objective function is the system resource multiplying the interference temperature. When two nodes are outside the PUs' transmission range and are in each other's transmission, they are allowed to communicate with each other, according to the transmission range and the interference range defined in Section 4.2.2.

The centralized problem is essentially linear programming, and we use the MATLAB built-in function "linprog" to solve the problem. After solving the centralized optimization and achieving the whole network topology, we can select the routes for given source and destination pairs, minimizing the aggregate interference to the PUs. However, we need a tradeoff between the cumulative delay and the aggregate interference. Therefore, given the constraint in Eq. (4.10), the source will then select the route with the lowest aggregate interference to the primary users.

4.4 Simulation Results and Discussions

In this section, simulation results for the network formation game based distributed routing algorithm for CR-WMNs are presented. We consider that the nodes are randomly deployed in a 250-by-250 meter area. The value of path loss constant is 2. We assume that link capacities only depend on the distance between the two players to simplify the problem. Therefore, the data rate is 48 Mbps within the distance of 32m , 36 Mbps within 37m, 24 Mbps within 45m, 18 Mbps within 60m, 12 Mbps within 69m, 9 Mbps within 77m, and 6 Mbps within 90m [92]. The transmission range R_T is 90m and the interference range R_I is 180m. The number of the nodes in the network may change in different simulations, and we consider random topologies. We generate a data set of 1,000 for the simulation. For every data set, the generated traffic by the node, the locations of the gateway are randomly generated and defined.

Figure 4.2 shows the simulation results for the proposed distributed routing algorithm. We use a random priority in Phase 1 of Algorithm 4.1 for a general case. The small black dots represent 50 SUs, while the big dot is a PU with the sector area as its footprint. All the SUs that are inside

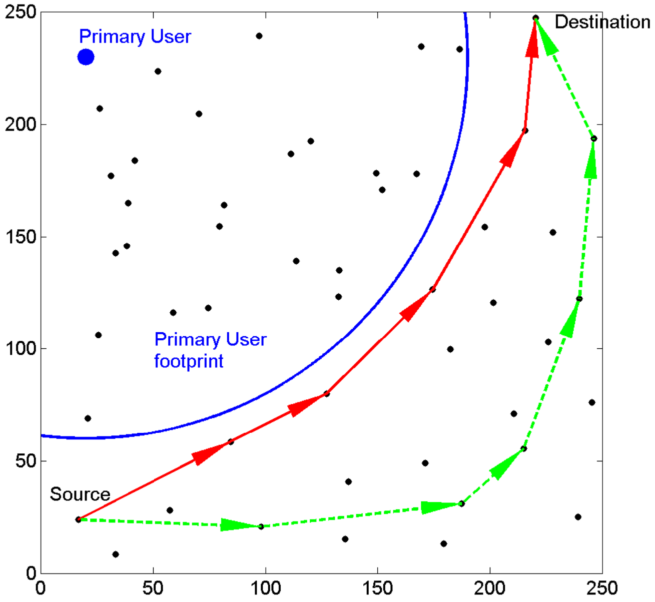

Figure 4.2 A simulation result after applying the proposed distributed algorithm in a 250-by-250 meter area.

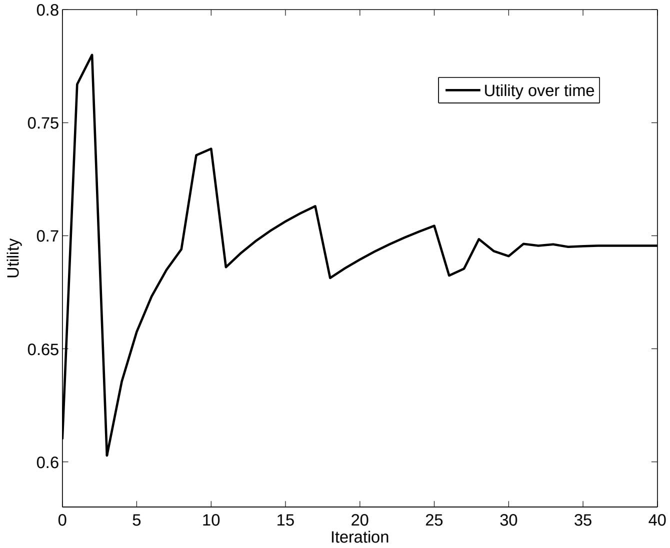

Figure 4.3 Utility over time.

the PU's footprint are forced to turn off because the spectrum is occupied by the PU. We also define the source and destination nodes in Figure 4.2. The route shown as the dashed arrows is the result after applying the proposed distributed routing algorithm. If we use the Dijkstra's shortest

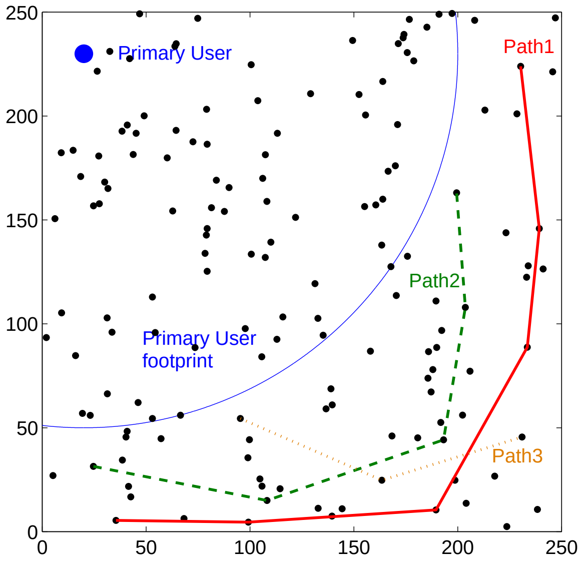

Figure 4.4 A simulation result for multiple source-destination pairs using distributed optimization in a 250-by-250 meter area.

path routing algorithm without considering the aggregate interference from the SUs to the PU, the route is achieved shown as the solid route. In these two routes, the interference temperature values to the PU are 1.6195 and 1.3354, respectively. Obviously, since the nodes in the solid route are closer to the PU than those in the dashed route, the solid route produces higher interference to the PU than the dashed route. In addition, the simulation results of the convergence of a selected node is shown in Figure 4.3, and we can find that the graph converges in about 35 loops based on this simulation setup. Therefore, the computation complexity of the proposed distributed algorithm should be $O(N)$, because the computation complexity of reach round in the algorithm is proportional to the number of nodes in the network. In addition, Figure 4.4 shows the simulation results for 3 source-destination pairs, and there are totally 150 secondary nodes randomly deployed in the 250-by-250-meter-area network. The solid route represents path1, the green dashed line represents path 2, and the other yellow dashed line is path 3. Simulation results show that the source nodes try to find the routes that are farther away from the PU, avoiding high aggregate interference to the PU.

In the next step, we compare the proposed distributed algorithm and the upper bound. Figure 4.5 shows the simulation results about the interference comparison given different numbers of the

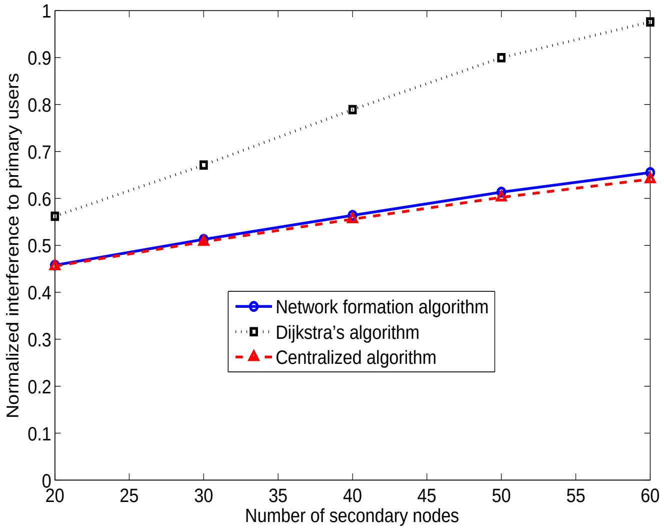

Figure 4.5 Interference given different numbers of secondary nodes in centralized and distributed algorithms.

SUs. The y-axis is the interference to the PU, which is normalized by the cumulative interference of 5 SUs that are R_T away from the PU, and the x-axis represents the number of the SUs. We fix the PU's location in this simulation as shown in Figure 4.2. ε_1, ε_2 are both set to be 1.5, and $\kappa 1$, $\kappa 2$ are 0.01. Small κ values can avoid the cost function changing too fast. Delay threshold is set to be twice the delay if using the Dijkstra's algorithm. The solid line represents the simulated performance of the distributed network formation algorithm. The dashed line is the result after the centralized solution, and it performs better than the distributed approach as expected. In addition, we can find that the distributed solution is very close to the optimum compared with the upper bound, producing about 1.0098 time the interference from the centralized algorithm. This means that it is 99.02$\%$ efficient compared to the upper bound. The black dashed line is the result using the Dijkstra's shortest path routing algorithm without considering the aggregate interference to the PUs. We can find that it produces the highest interference in the three solutions. Moreover, with the increasing number of SUs, interference to the PUs increases in Figure 4.5. Note that the reason that we only compare the proposed algorithms with the Dijkstra's shortest path algorithm is that most other existing routing algorithms for CR networks do not consider the aggregate interference to the PUs.

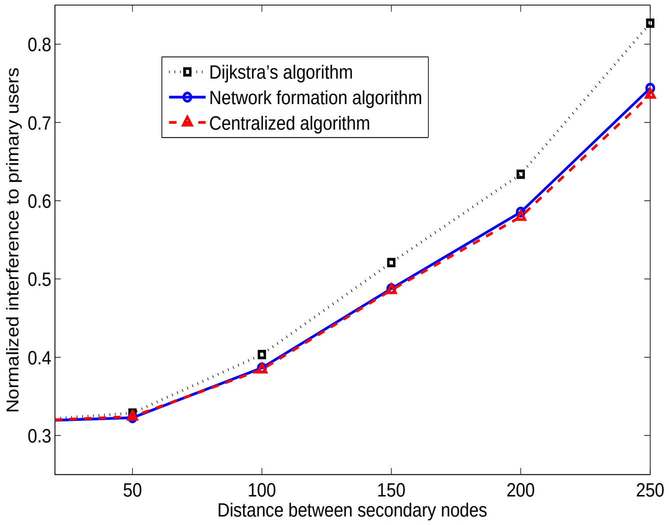

Figure 4.6 Interference given different distances between secondary nodes in centralized and distributed algorithms.

In Figure 4.6, another comparison between the distributed and centralized algorithms is provided. The y-axis is the interference to the PUs, and the x-axis represents the distance between different SUs in meters. There are 150 secondary users randomly deployed in this simulation. The results are similar as in Figure 4.5. In addition, the interference to the PUs increases with the increasing of x-axis in Figure 4.6. This is because the route between two SUs will include more nodes and more hops, which means more total interference to the PU, when the distance between two SUs increases.

In Figure 4.7 and Figure 4.8, the comparison of delay between the proposed distributed algorithm and the centralized algorithm is presented. The y-axis represents the total delay along the route, and the x-axis represents the number of SUs and the distance between SUs, respectively. For simplicity, the delay is defined as the number of hops. Firstly, we can find that with the increasing of the number of SUs and the increasing of the distance between SUs, the total delay will increase, which is consistent with the results in Figure 4.5 and Figure 4.6. Secondly, the centralized approach provides a slightly higher delay than the distributed network formation algorithm since the network formation algorithm provides a slightly higher interference than the centralized algorithm. In addition, the Dijkstra's algorithm performs the best, and it always finds the shortest path without

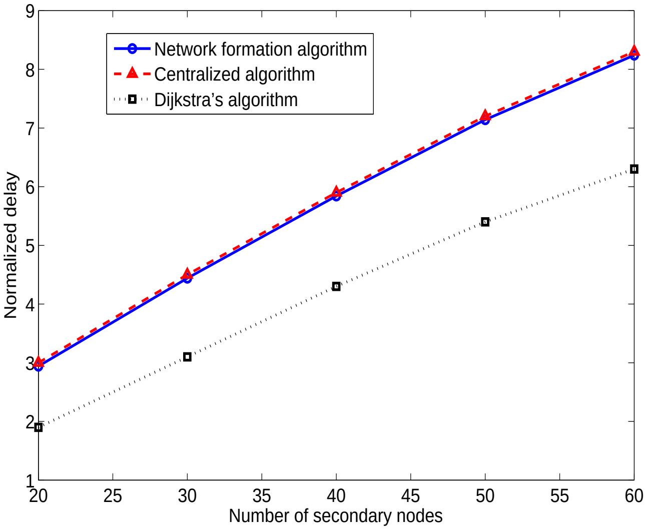

Figure 4.7 Delay given different numbers of secondary nodes in centralized and distributed algorithms.

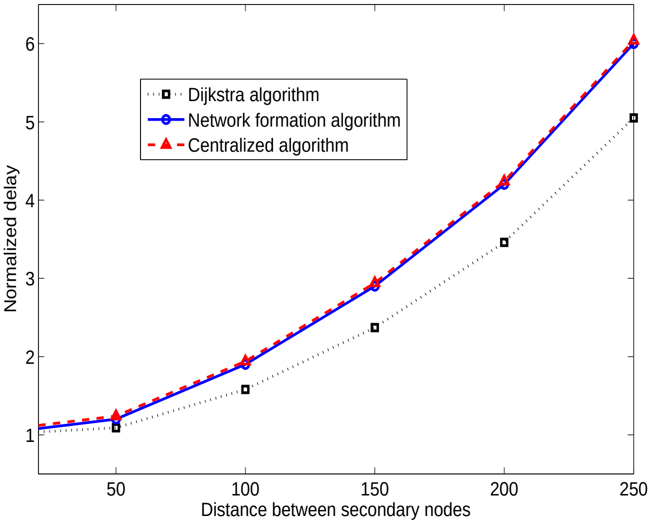

Figure 4.8 Delay given different distances between secondary nodes in centralized and distributed algorithms.

considering the aggregate interference to the PUs.

If we do not set a delay threshold shown in (4.10) in a large area with a significantly large amount of secondary nodes, the route will be very long with large delay, although the aggregate interference to the primary users is decreased. This is not acceptable and we need to use delay threshold to constrain the route. In Figure 4.9 and Figure 4.10, we show the performance comparison

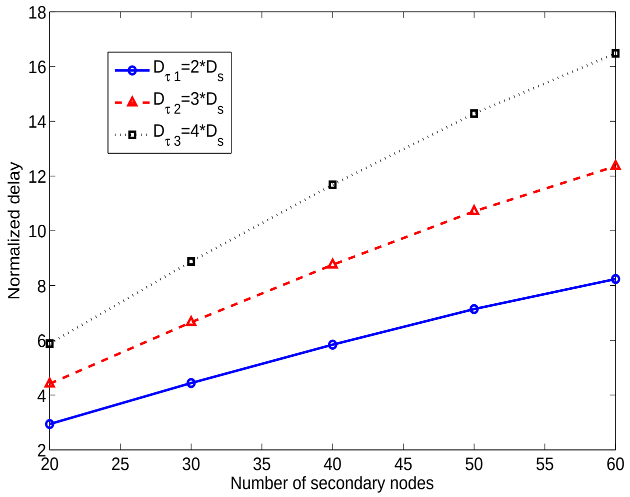

Figure 4.9 Comparison of delay for different delay thresholds.

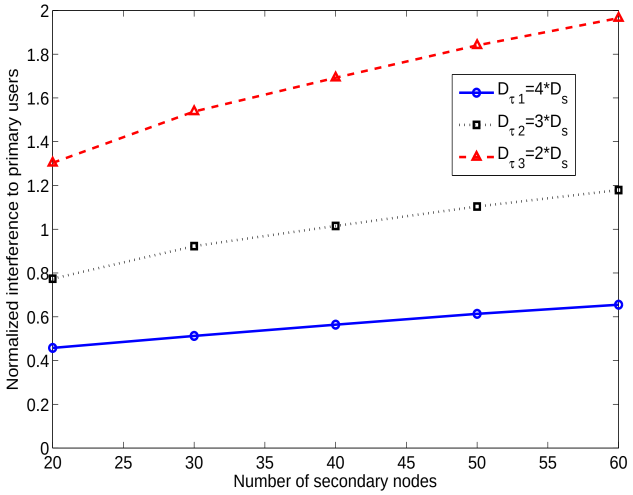

Figure 4.10 Comparison of aggregate interference given different delay thresholds.

between different delay thresholds using the distributed network formation algorithm. D_s represents the delay of the route between the source and destination using the Dijkstra's algorithm. We can find in Figure 4.9 and Figure 4.10 that higher delay threshold provides longer path and lower aggregate interference to the primary user. With a higher delay threshold, although the path we find is longer with more secondary nodes and farther away from the primary user, the aggregate interference decreases exponentially with distance, which is much faster than the linear increasing of number of nodes on the route.

4.5 Conclusion

In this chapter, we developed a distributed routing algorithm in the cognitive radio based wireless mesh networks using network formation game. In CR-WMNs, a single SU may produce low interference to the PUs if the distance between itself and the PUs is big enough. Nevertheless, the aggregate interference from a large number of SUs transmitting at the same time can be significant, which will influence the primary users' transmissions. Therefore, we developed a distributed routing algorithm using the network formation game framework to minimize the cumulative interference to the PUs, which is practically implementable. From the simulation results, we can see that the proposed algorithm can provide the routes with low aggregate interference to the PUs compared to the Dijkstra's algorithm. We also compared the performance of the distributed optimization algorithm with an upper bound and validated its efficiency, and the distributed solution is very close to the optimum, providing 99.02% efficiency of the upper bound. Future work will include deriving a mathematical model using price of anarchy in the game theory to analyze the ratio of the distributed solution to the optimal solution.

Chapter 5

Conclusion and Future Work

5.1 Summary and Conclusion

In this dissertation, efforts have been made to solve the routing problems in cognitive radio networks and the security problems in cognitive radio systems. The techniques that have been developed in this dissertation are listed as follows:

- A belief-propagation based defense strategy for the primary user emulation attack in cognitive radio networks is proposed, which can provide low computational complexity and fast execution speed. Also our proposed defense framework can avoid deploying the expensive hardware, and is flexible to incorporate to defeat various kinds of attacks for future extension.
- A brand new network-layer attack, named routing toward primary user attack, is discovered, in which malicious secondary users will try to route the data to those secondary users which are closer to the primary users in order to increase interference to the primary users. This new attack is very hard to detect. Also a belief-propagation based defense algorithm is proposed which can effectively detect the malicious users.
- An interference minimization routing algorithm using network formation game for the cognitive radio networks is proposed, which is able to minimize the aggregate interference from the secondary users to the primary users while keeping the delay along the route low. The proposed distributed routing algorithm is practically implementable and the performance is very close to the optimum.

The proposed techniques can provide a lot of benefits to society. Modern society depends on wireless networks in order to facilitate ubiquitous access to the Internet, other human users, and both essential services and modern conveniences. Therefore, conducting research into cognitive radio wireless networks, which could be employed in many emerging applications and facilitate the usage of the wireless spectrum, is important. The routing techniques and technology concerning the security issues in cognitive radio networks will also assist the public safety, emergency services, and first responders communities in enabling better communications access to the network, which could potentially translate into additional human lives being saved. Furthermore, the proposed technology can also be employed in commercial data networking devices, such as vehicular communication networks, in order to further enhance the quality-of-life through ubiquitous wireless access.

5.2 Future Work

There exists a number of areas for future work related to what has been presented in this dissertation.

- *Extend for cross-layer defense for various attacks.* Each defense strategy proposed in this dissertation is developed to defeat a certain attack in cognitive radio systems. In the future, we could design a cross-layer defense algorithm using the framework of belief propagation to defeat various attacks in various layers. We can design a new local function and a new compatibility function or combine the original local functions and compatibility functions. Also weighting between different attacks in the defense algorithm can be optimized to fit different scenarios.

- *Collaborative spectrum sensing using a space-time Kalman filter.* Collaborative spectrum sensing between several secondary users can greatly increase the sensing speed and the detection accuracy in cognitive radio systems. Specifically, we will consider vehicular networks, and apply the space-time Kalman filter to estimate the primary users' signal strength along the high way. In this case, each vehicle is equipped with cognitive radio, and they are able to detect the primary users' signals. This information can be used for the dynamic spectrum access over the vehicular cognitive radio net-works. Due to the fact that individual vehicle has limited view for the spectrum over different places, we further propose the gossip and distributed Kalman filtering, where the different cognitive radios in different vehicles can collaborate with each other. When two vehicles are close to each other on the highway, they can choose to exchange the estimations, and this can help to improve the detection and estimation efficiency.

- *Incorporate MAC layer consideration.* The distributed routing algorithm proposed in this dissertation only considers the network layer. In the future, we could incorporate the MAC layer and develop a cross-layer routing algorithm. However, the network formation game might not be a good choice, since it is hard to include the scheduling in the utility function while keeping the distributed feature. We may need to find a new game model to develop the cross-layer routing algorithm.

- *Incorporate cognitive pilot channel and medial axis.* Cognitive pilot channel (CPC) can be employed so that secondary users are able to access the information about the primary users, including primary users' locations and channel conditions. Also medial axis with a relaxation factor can be used as a reference path for the routing, along which we can develop a hierarchical structure for multiple sources to reach their destinations. The points on the medial axis provide the lowest interference to the primary users, and the routes can be established which are very close to the medial axis within the limit of the relaxation factor.

- *Extend for dynamic game framework for routing.* The network formation game framework proposed in this dissertation is only deterministic. We can construct a temporal and spatial dynamic non-cooperative game to model the interactions among the secondary users as well as their influences on the primary users. A multi-stage fictitious play learning can also be employed for the distributed routing in the multi-hop cognitive radio networks. We can obtain a set of mixed (behavioral) Nash equilibrium strategies of the dynamic game in a closed form by backward induction. The proposed algorithm can minimize the overall interference from the secondary users to the primary users, as well as the average packet delay along the route from the secondary nodes to their destinations.

- *Real-world implementation and test.* It would be interesting to have the proposed techniques implemented on a cognitive /software-defined radio hardware platform. The Universal Software Radio Peripherals 2 (USRP2) in the Wireless Networking, Signal Processing and Security Laboratory at the University of Houston, which is a high-speed USB-based board for making software-defined radios, can be a good candidate for implementation of the proposed techniques. GNU Radio is an open source toolkit that can be used to develop software-defined radios. This framework uses a combination of C++ and Python to optimize DSP performance while providing an easy-to-use application programming environment.

Bibliography

[1] E. Hossain, D. Niyato, and Z. Han, Eds., *Dynamic spectrum access and management in cognitive radio networks*. Cambridge University Press, June 2009.

[2] Z. Yuan, “Sidelobe suppression and agile transmission techniques for multicarrier-based cognitive radio systems,” Master’s thesis, Electrical and Computer Engineering, Worcester Polytechnic Institute, Worcester, MA, USA, May 2009.

[3] “Report of the spectrum efficiency group,” FCC Spectrum Policy Task Force, Tech. Rep., 2002.

[4] M. A. McHenry, P. A. Tenhula, D. McCloskey, D. A. Roberson, and C. S. Hood, “Chicago spectrum occupancy measurements & analysis and a long-term studies proposal,” in *Proc. of First International Workshop on Technology and Policy for Accessing Spectrum (TAPAS)*, New York, NY, USA, August 2006, p. 1.

[5] R. V. Prasad, P. Pawlczak, J. A. Hoffmeyer, and H. S. Berger, “Cognitive functionality in next generation wireless networks: Standardization efforts,” *IEEE Communications Magazine*, vol. 46, no. 4, p. 72, April 2008.

[6] Q. Zhao and B. M. Sadler, “A survey of dynamic spectrum access,” *IEEE Signal Processing Magazine*, vol. 24, no. 3, p. 79, May 2007.

[7] W. Lehr and J. Crowcroft, “Managing shared access to a spectrum commons,” in *Proc. of IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, Baltimore, MD, USA, November 2005, pp. 420–444.

[8] D. Hatfield and P. Weiser, “Property rights in spectrum: Taking the next step,” in *Proc. of IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, Baltimore, MD, USA, November 2005, pp. 43–55.

[9] V. D. Chakravarthy, Z. Wu, A. Shaw, M. A. Temple, R. Kannan, and F. Garber, “A general overlay/underlay analytic expression representing cognitive radio waveform,” in *Proc. of Waveform Diversity and Design Conference*, 2007.

[10] F. K. Jondral, “Software-defined radio - basics and evolution to cognitive radio,” *EURASIP Journal on Wireless Communications and Networking*, no. 3, pp. 275–283, 2005.

[11] J. M. III, “Cognitive radio for flexible mobile multimedia communications,” in *Proc. of IEEE International Workshop on Mobile Multimedia Communications*, November 1999.

[12] I. F. Akyildiz, W. Lee, M. C. Vuran, and S. Mohanty, “Next generation/dynamic spectrum access/cognitive radio wireless networks: A survey,” *Elsvier Computer Networks Journal*, vol. 50, pp. 2127–2159, May 2006.

[13] S. Haykin, “Cognitive radio: Brain-empowered wireless communications,” *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 2, pp. 201–220, Febuary 2005.

[14] S. J. Shellhammer, A. K. Sadek, and W. Zhang, “Technical challenges for cognitive radio in the tv white space spectrum,” in *Proc. of 2009 Information Theory and Applications Workshop*, San Diego, CA, USA, Febuary 2009.

[15] D. Cabric, S. M. Mishra, and R. W. Brodersen, “Implementation issues in spectrum sensing for cognitive radios,” in *Proc. of IEEE Asilimar Conference on Signals, Systems and Computers*, Pacific Grove, CA, US, November 2004, pp. 772–776.

[16] K. R. Chowdhury and I. F. Akyildiz, “Cognitive wireless mesh networks with dynamic spectrum access,” *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 1, pp. 168 – 181, January 2008.

[17] N. Nie and C. Comaniciu, “Adaptive channel allocation spectrum etiquette for cognitive radio networks,” in *Proc. of IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks*, Baltimore, MD, US, November 2005, pp. 269–278.

[18] O. Ileri, D. Samardzija, T. Sizer, and N. B. Mandayam, “Demand responsive pricing and competitive spectrum allocation via a spectrum server,” in *Proc. of IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks*, Baltimore, MD, US, November 2005, pp. 194–202.

[19] R. Etkin, A. Parekh, and D. Tse, “Spectrum sharing for unlicensed bands,” in *Proc. of IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks*, Baltimore, MD, US, November 2005, pp. 251–258.

[20] D. I. Kim, L. B. Le, and E. Hossain, “Joint rate and power allocation for cognitive radios in dynamic spectrum access environment,” *IEEE Transactions on Wireless Communications*, vol. 7, no. 12, pp. 5517–5527, December 2008.

[21] Cognitive radio tutorial, available from http://www.radio-electronics.com.

[22] T. Yucek and H. Arslan, “A survey of spectrum sensing algorithms for cognitive radio applications,” *IEEE Communications Surveys & Tutorials*, vol. 11, no. 1, pp. 116–130, 2009.

[23] K. J. R. Liu and B. Wang, Eds., *Cognitive radio networking and security: A game-theoretic view*. Cambridge University Press, November 2010.

[24] C. Xin, B. Xie, and C. C. Shen, “A novel layered graph model for topology formation and routing in dynamic spectrum access networks,” in *Proc. of First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN’05)*, November 2005, p. 308C317.

[25] B. Zhang, Y. Takizawa, A. Hasagawa, A. Yamaguchi, and S. Obana, “Tree-based routing protocol for cognitive wireless access networks,” in *Proc. of IEEE Wireless Communications and Networking Conference*, Kowloon, Hong Kong, China, March 2007.

[26] C. W. Pyo and M. Hasegawa, “Minimum weight routing based on a common link control radio for cognitive wireless ad hoc networks,” in *Proc. of the 2007 International Conference on Wireless Communications and Mobile Computing (IWCMC07)*, Agust 2007, p. 399C404.

[27] Y. Shi and Y. Hou, "A distributed optimization algorithm for multi-hop cognitive radio networks," in *Proc. of the 27th IEEE Conference on Computer Communications (INFOCOM'08)*, March 2008, p. 1292C1300.

[28] G. Cheng, W. Liu, Y. Li, and W. Cheng, "Spectrum aware on-demand routing in cognitive radio networks," in *Proc. of the 2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN'07)*, April 2007, p. 571C574.

[29] M. Cesana, F. Cuomo, and E. Ekici, "Routing in cognitive radio networks: Challenges and solutions," accepted: June 30, 2010, preprint, available at http://www2.ece.ohio-state.edu/~ekici/papers/crnroutingsurvey.pdf.

[30] I. Pefkianakis, S. Wong, and S. Lu, "Samer: spectrum aware mesh routing in cognitive radio networks," in *Proc. of the 3rd IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN'08)*, October 2008, pp. 1–5.

[31] L. Ding, T. Melodia, S. N. Batalama, J. D. Matyjas, and M. J. Medley, "Corss-layer routing and dynamic spectrum allocation in cognitive radio ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 4, pp. 1969–1979, May 2010.

[32] A. Abbagnale and F. Cuomo, "Gymkhana: A connectivity-based routing scheme for cognitive radio ad hoc networks," in *Proc. of IEEE Conference on Computer Communications Workshops*, San Diego, CA, US, May 2010.

[33] I. Filippini, E. Ekici, and M. Cesana, "Minimum maintenance cost routing in cognitive radio networks," in *Proc. of the IEEE 6th International Conference on Mobile Adhoc and Sensor Systems (MASS'09)*, October 2009, p. 284C293.

[34] Q. Wang and H. Zheng, "Route and spectrum selection in dynamic spectrum networks," in *Proc. of 3rd IEEE Consumer Communications and Networking Conference (CCNC)*, vol. 1, January 2006.

[35] C. Xin, B. Xie, and C. C. Shen, “A novel layered graph model for topology formation and routing in dynamic spectrum access networks,” in *Proc. of First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN 2005)*, vol. 1, November 2005, pp. 308–317.

[36] Q. Guan, F. R. Yu, and S. Jiang, “Prediction-based topology control and routing in cognitive radio mobile ad hoc networks,” in *Proc. of IEEE Conference on Computer Communications Workshops*, San Diego, CA, US, March 2010.

[37] B. Li, D. Li, Q. Wu, and H. Li, “Asar: Ant based spectrum routing for cognitive radio networks,” in *Proc. of International Conference on Wireless Communication & Signal Processing*, Nanjing, China, November 2009, pp. 1–5.

[38] Z. Song, B. Shen, Z. Zhou, and K. S. Kwak, “Improved ant routing algorithm in cognitive radio networks,” in *Proc. of 9th International Symposium on Communications and Information Technology*, Icheon, South Korea, September 2009, pp. 110–114.

[39] Q. He and H. Zhou, “Research on the routing algorithm based on qos requirement for cognitive radio networks,” in *Proc. of International Conference on Computer Science and Software Engineering*, Wuhan, Hubei, China, December 2008.

[40] W. Jiang, H. Cui, and J. Chen, “Spectrum-aware cluster-based routing protocol for multiple-hop cognitive wireless network,” in *Proc. of IEEE International Conference on Communications Technology and Applications*, Beijing, China, October 2009.

[41] A. G. Fragkiadakis, E. Z. Tragos, and I. G. Askoxylakis, “A survey on security threats and detection techniques in cognitive radio networks,” accepted for publication in *IEEE COMMUNICATIONS SURVEYS & TUTORIALS*.

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

[43] Z. Jin, S. Anand, and K. P. Subbalakshmi, “Detecting primary user emulation attacks in dynamic spectrum access networks,” in *Proc. of IEEE International Conference on Communication (ICC’09)*, June 2009.

[44] H. Li and Z. Han, “Dogfight in spectrum: Jamming and anti-jamming in multichannel cognitive radio systems,” in *Proc. of IEEE Global Communication Conference (Globecom’09)*, November 2009.

[45] R. Chen, J. M. Park, and K. Bian, “Robust distributed spectrum sensing in cognitive radio networks,” in *Proc. of the 27th Conference on Computer Communications (Infocom’08)*, April 2008.

[46] P. Kaligineedi, M. Khabbazian, and V. Bhargava, “Secure cooperative sensing techniques for cognitive radio systems,” in *Proc. of IEEE International Conference on Communications (ICC’08)*, May 2008.

[47] W. Wang, H. Li, Y. Sun, and Z. Han, “Attack-proof collaborative spectrum sensing in cognitive radio systems,” in *Proc. of Conference on Information Sciences and Systems (CISS’09)*, March 2009.

[48] W. Wang, H. Li, Z. Han, and Y. Sun, “Catchit: Detect malicious nodes in collaborative spectrum sensing,” in *Proc. of IEEE Global Communication Conference (Globecom’09)*, November 2009.

[49] K. Bian and J. M. Park, “Mac-layer misbehaviors in multi-hop cognitive radio networks,” in *Proc. of 2006 US - Korea Conference on Science, Technology, and Entrepreneurship (UKC2006)*, August 2006.

[50] G. Jakimoski and K. Subbalakshmi, “Denial-of-service attacks on dynamic spectrum access networks,” in *Proc. of IEEE International Conference on Communications Workshops (ICC Workshops’08)*, May 2008.

[51] S. Kurosawa1, H. Nakayama1, N. Kato1, A. Jamalipour2, , and Y. Nemoto1, “Detecting blackhole attack on aodv-based mobile ad hoc networks by dynamic learning method,” *International Journal of Network Security*, vol. 5, pp. 338–346, 2007.

[52] R. Maheshwari, J. Gao, and S. R. Das, “Detecting wormhole attacks in wireless networks using connectivity information,” in *Proc. of 26th IEEE International Conference on Computer Communications (Infocom’07)*, May 2007.

[53] J. R. Douceur, “The sybil attack,” in *Proc. of the 1st International Workshop on Peer-to-Peer Systems (IPTPS ’02)*, March 2002.

[54] T. Zhou, R. R. Choudhury, P. Ning, and K. Chakrabarty, “P2dap - sybil attacks detection in vehicular ad hoc networks,” *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 3, pp. 582–594, March 2011.

[55] Y. Hu, A. Perrig, and D. Johnson, “Packet leashes: A defense against wormhole attacks in wireless ad hoc networks,” in *Proc. of 22nd Annual Joint Conference of the IEEE Computer and Communications Societies (Infocom’03)*, March 2003.

[56] D. M. Shila and T. Anjali, “A game theoretic approach to gray hole attacks in wireless mesh networks,” in *Proc. of Military Communications Conference (Milcom’08)*, November 2008.

[57] H. Sun, J. C. S. Lui, and D. K. Y. Yau, “Defending against low-rate tcp attacks: dynamic detection and protection,” in *Proc. of the 12th IEEE International Conference on Network Protocols (ICNP’04)*, October 2004.

[58] B. Yu and B. Xiao, “Detecting selective forwarding attacks in wireless sensor networks,” in *Proc. of 20th International Parallel and Distributed Processing Symposium (IPDPS’06)*, April 2006.

[59] J. S. Yedidia, W. T. Freeman, and Y. Weiss, Eds., *Understanding belief propagation and its generalizations in exploring artificial intelligence in the new millennium.* Science and Technology Books, 2003, chap. 8, pp. 2282-2312.

[60] S. Anand, Z. Jin, and K. P. Subbalakshmi, “An analytical model for primary user emulation attacks in cognitive radio networks,” in *Proc. of IEEE Symposia on New Frontiers in Dynamic Spectrum Access Networks (DySPAN’08)*, Chicago, IL, US, October 2008.

[61] L. Huang, L. Xie, H. Yu, W. Wang, and Y. Yao, “Anti-pue attack based on joint position verification in cognitive radio networks,” in *Proc. of 2010 International Conference on Communications and Mobile Computing (CMC)*, Shenzhen, Guangdong, China, April 2010.

[62] Z. Chen, T. Cooklev, C. Chen, and C. Pomalaza-Raez, “Modeling primary user emulation attacks and defenses in cognitive radio networks,” in *Proc. of Performance Computing and Communications Conference (IPCCC)*, Scottsdale, AZ, December 2009.

[63] R. W. Thomas, R. S. Komali, B. J. Borghetti, and P. Mahonen, “A bayesian game analysis of emulation attacks in dynamic spectrum access networks,” in *Proc. of 2010 IEEE Symposium on New Frontiers in Dynamic Spectrum*, April 2010.

[64] Y. Liu, P. Ning, and H. Dai, “Authenticating primary users’ signals in cognitive radio networks via integrated cryptographic and wireless link signatures,” in *Proc. of 2010 IEEE Symposium on Security and Privacy*, Oakland, May 2010.

[65] X. Tan, K. Borle, W. Du, and B. Chen, “Cryptographic link signatures for spectrum usage authentication in cognitive radio,” in *Proc. of Fourth ACM conference on Wireless network security*, Hamburg, Germany, June 2011.

[66] Z. Yuan, D. Niyato, H. Li, and Z. Han, “Defense against primary user emulation attacks using belief propagation of location information in cognitive radio networks,” in *Proc. of IEEE Wireless Communications and Networking Conference (WCNC’11)*, Cancun, Mexico, March 2011.

[67] H. Li, “Cooperative spectrum sensing via belief propagation in spectrum-heterogeneous cognitive radio systems,” in *Proc. of IEEE Wireless Communications and Networking Conference (WCNC’10)*, Sydney, Australia, April 2010.

[68] S. M. Ross, Ed., *Introduction to probability models, ninth edition*. Academic Press, 2007.

[69] B. Wild and K. Ramchandran, “Detecting primary receivers for cognitive radio applications,” in *Proc. of IEEE Dynamic Spectrum Access Networks*, November 2005.

[70] T. Roos, P. Myllymaki, and H. Tirri, “A statistical modeling approach to location estimation,” *IEEE Transactions on Mobile Computing*, vol. 1, no. 1, pp. 59–69, January 2002.

[71] A. T. Ihler, J. W. Fisher, R. L. Moses, and A. S. Willsky, “Nonparametric belief propagation for self-localization of sensor networks,” *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 4, pp. 809–819, April 2005.

[72] H. Li and D. K. Irick, “Collaborative spectrum sensing in cognitive radio vehicular ad hoc networks: Belief propagation on highway,” in *Proc. of IEEE Vehicle Technology Conference (VTC’10)*, May 2010.

[73] J. M. Mooij and H. J. Kappen, “Sufficient conditions for convergence of the sum-product algorithm,” *IEEE Transactions on Information Theory*, vol. 3, no. 12, pp. 4422–4437, December 2007.

[74] J. Dieudonne, “Foundations of modern analysis,” *ser. Pure and Applied Mathematics*, vol. 10, no. 1, 1969.

[75] C. M. Gruner and D. H. Johnson, “Calculation of the kullback-leibler distance between point process models,” in *Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing*, Salt Lake City, UT, US, 2001.

[76] G. Ferrari and G. Corazza, “Tight bounds and accurate approximations for dqpsk transmission bit error rate,” *ELECTRONICS LETTERS*, vol. 40, no. 20, September 2004.

[77] R. D. Taranto, H. Yomo, P. Popovski, K. Nishimori, and R. Prasad, “Cognitive mesh network under interference from primary user,” *Wireless Personal Communications*, vol. 45, no. 3, pp. 385–401, May 2008.

[78] A. Goldsmith, Ed., *Wireless communications*. Cambridge University Press, 2005.

[79] Y. Yuan, P. Bahl, R. Chandra, T. Moscibroda, and Y. Wu, "Allocating dynamic time-spectrum blocks in cognitive radio networks," in *Proc. of the 8th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, Montreal, Canada, September 2007, pp. 130–139.

[80] R. C. Pereira, R. D. Souza, and M. E. Pellenz, "Using cognitive radio for improving the capacity of wireless mesh networks," in *Proc. of IEEE Vehicular Technology Conference*, Calgary, Canada, September 2008.

[81] X. Chen, K. Makki, K. Yen, and N. Pissinou, "Sensor network security: A survey," *IEEE Communications Surveys and Tutorials*, vol. 11, pp. 52–73, June 2009.

[82] A. K. Sadek, Z. Han, and K. J. R. Liu, "Distributed relay-assignment protocols for coverage expansion in cooperative wireless networks," *IEEE Transactions on Mobile Computing*, vol. 9, no. 4, pp. 505–515, April 2010.

[83] Z. Yuan, J. B. Song, and Z. Han, "Interference minimization routing and scheduling in cognitive radio wireless mesh networks," in *Proc. of IEEE Wireless Communications and Networking Conference*, April 2010.

[84] B. Frey, Ed., *Graphical models for machine learning and digital communications*. MIT Press, 1998.

[85] Z. Han and Y. Sun, "Securing cooperative transmission in wireless communications," in *Proc. of First Workshop on the Security and Privacy of Emerging Ubiquitous Communication Systems*, Pennsylvania, USA, August 2007.

[86] C. Dellarocas, "Mechanisms for coping with unfair ratings and discriminatory behavior in online reputation reporting systems," in *Proc. of 21st International Conference on Information Systems*, Brisbane, Queensland, Australia, December 2000.

[87] G. Parissidis, M. Karaliopoulos, T. Spyropoulos, and B. Plattner, “Interference-aware routing in wireless multihop networks,” *IEEE Transactions on Mobile Computing*, vol. 10, no. 5, pp. 716–733, May 2011.

[88] W. Saad, Z. Han, M. Debbah, A. Hjorungnes, and T. Basar, “A game-based self-organizing uplink tree for voip services in ieee 802.16j networks,” in *Proc. of IEEE International Conference on Communications*, Dresden, Germany, June 2009.

[89] T. Yucek and H. Arslan, “A survey of spectrum sensing algorithms for cognitive radio applications,” *IEEE Communications Surveys and Tutorials*, vol. 11, pp. 116–130, March 2009.

[90] T. C. Clancy, “Achievable capacity under the interference temperature model,” in *Proc. of IEEE International Conference on Computer Communications*, Anchorage, AK, US, May 2007, pp. 794–802.

[91] M. Yildiz, game theory lecture notes: Backward induction, available from hurkens.iae-csic.org/teaching/ideagametheory/lecture9.pdf.

[92] IEEE 802.11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications.