UNIVERSITY OF CALGARY

Stackelberg-Based Anti-Jamming Game for Cooperative Cognitive Radio Networks

by

Ismail Kamal Sayed Ahmed

A THESIS
SUBMITTED TO THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE
DEGREE OF DOCTOR OF PHILOSOPHY

GRADUATE PROGRAM IN ELECTRICAL AND COMPUTER ENGINEERING

CALGARY, ALBERTA
SEPTEMBER, 2017

© Ismail Kamal Sayed Ahmed 2017
Abstract

With a target to address the frequency spectrum scarcity, Cognitive Radio technology emerged as a solution to achieve enhanced spectrum utilization through enabling secondary users to opportunistically access the licensed frequency bands meant for the primary users. Cognitive Radio Networks (CRNs) are plagued with new security threats besides the traditional threats that are shared with other wireless networks. Primary security threats include the radio jammers who deliberately transmit radio signals to block, mask, or emulate the legitimate active wireless connections. Acute radio jammers only attack at CRNs’ vulnerable times to cause maximum damage while saving power and decreasing the probability of being detected.

In this thesis, using the IEEE 802.22 CRNs as a basis, a security threat assessment is conducted, and a deception-based Stackelberg game anti-jamming mechanism is proposed. Unlike previous works in the literature, first, this thesis utilizes the Bayesian Attack Graph (BAG) model to facilitate the security assessment of CRNs, providing a feasible metric of CR vulnerabilities. Using the BAG model, the probability of denial of service in the IEEE 802.22 networks was proven to increase up to 51.3% when considering multiple attacks in comparison to the most severe sole attack.

Second, this thesis proposes a deception-based defense mechanism which aims at decreasing the contingent acute jamming attacks’ likelihood in targeting CRNs’ vulnerabilities. The Stackelberg framework is adopted to count for the bias in information which exists between the attacker and the defender due to the attacker’s reconnaissance capabilities. To this end, the Stackelberg equilibria between the attacker(s) and the defending CRN are calculated under the two cases when the players know and are uncertain about the primary user activity. Both theoretical analysis and numerical results show that the defending CRN can decrease the probability of success of the contingent acute jamming attacks when the
defender has the incentive to defend the channel.

Lastly, the thesis proves the usefulness of the proposed defense mechanism in the extreme case when the defender is uncertain about the attacker’s payoff function in a repeated game framework through online learning.
Acknowledgments

Firstly and above all, I would like to thank almighty God for his blessings throughout my research work and for the fulfillment of this doctoral endeavor.

I would like to thank my supervisor Dr. Abraham Fapojuwo for his endless guidance, fruitful discussions and non-fading encouragement. Working under his supervision helped me developing my research ability and improving my communication skills.

Last but not the least, I would like to thank my family, my mother, my father, my siblings and my lovely wife for supporting me spiritually throughout writing this thesis and my life in general. I would have not been able to make it this far without my family kindness, patience and endless support.
# Table of Contents

Abstract ................................................................. ii  
Acknowledgments ......................................................... iv  
Table of Contents ......................................................... v  
List of Symbols ............................................................ vii  
1 Introduction ............................................................. 1  
1.1 Context and Background ............................................... 1  
1.2 Cognitive Radio Security Challenges and Opportunities ................. 4  
1.3 Problem Statement and Thesis Objectives .................................. 7  
1.4 Contributions and Outline ............................................... 8  
2 Related Work ............................................................ 11  
2.1 Security Threat Assessment of the IEEE 802.22 CRN ...................... 12  
2.2 Deceiving the Deceivers in CRNs ....................................... 14  
2.3 Learning in Repeated Games ............................................ 18  
3 Security Threat Assessment of The  
IEEE 802.22 CRNs .................................................... 23  
3.1 Introduction ........................................................... 23  
3.1.1 Impairment of Sensory Information .................................. 24  
3.1.2 Location Falsification and Location Failure .......................... 24  
3.1.3 Control Channel Failure ............................................ 25  
3.1.4 Databases Failure .................................................... 25  
3.1.5 Deception of Databases ............................................. 26  
3.1.6 Spurious Operating System Commands ................................ 26  
3.2 The Bayesian Attack Graph (BAG) Model ................................ 26  
3.2.1 BAG Model’s Main Components ...................................... 26  
3.2.2 Unconditional Probability of Attacking a Node ....................... 29  
3.3 DoS Threat Assessment of the IEEE 802.22 Networks Using BAG Model . 32  
3.3.1 BAG Model Assumptions ............................................. 32  
3.3.2 Building of DoS AG model .......................................... 33  
3.3.3 Building of DoS BAG model ......................................... 33  
3.3.4 Simultaneous Multiple Attack Scenarios ............................. 35  
3.3.5 Simulation Results and Interpretation ............................... 36  
3.4 Chapter Summary ....................................................... 40  
4 Deceiving The Deceiver in CR networks .................................. 41  
4.1 Introduction ........................................................... 41  
4.2 System Model .......................................................... 42  
4.2.1 The Attacker Model .................................................. 42  
4.2.2 The Defender Model ................................................ 44  
4.2.3 The Payoff Functions and the Normal Form .......................... 47  
4.3 The IEEE 802.22 Stackelberg Deception-based Game Problem ........... 51  
4.3.1 Guesstimating Game Parameters ................................... 52  
4.3.2 The Special Case: A Game with Complete Information ............. 58  
4.3.3 The General Case: A Game with Incomplete Information .......... 66
## List of Symbols, Abbreviations and Nomenclature

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>U of C</td>
<td>University of Calgary</td>
</tr>
<tr>
<td>CR</td>
<td>Cognitive Radio</td>
</tr>
<tr>
<td>CRN</td>
<td>Cognitive Radio Network</td>
</tr>
<tr>
<td>PU</td>
<td>Primary User</td>
</tr>
<tr>
<td>SU</td>
<td>Secondary User</td>
</tr>
<tr>
<td>RF</td>
<td>Radio Frequency</td>
</tr>
<tr>
<td>WRAN</td>
<td>Wireless Regional Access Network</td>
</tr>
<tr>
<td>CBS</td>
<td>Cognitive Base Station</td>
</tr>
<tr>
<td>SDR</td>
<td>Software Defined Radio</td>
</tr>
<tr>
<td>TV</td>
<td>Television</td>
</tr>
<tr>
<td>QP</td>
<td>Quiet Period</td>
</tr>
<tr>
<td>PMP</td>
<td>Point–to–Multi–Point</td>
</tr>
<tr>
<td>CPEs</td>
<td>Customer Premises Equipments</td>
</tr>
<tr>
<td>IEEE</td>
<td>Institute of Electrical and Electronics Engineers</td>
</tr>
<tr>
<td>DoS</td>
<td>Denial of Service</td>
</tr>
<tr>
<td>PUE</td>
<td>Common Control Channel</td>
</tr>
<tr>
<td>CCC</td>
<td>Television</td>
</tr>
<tr>
<td>DARPA</td>
<td>The Defense Advanced Research Projects Agency</td>
</tr>
<tr>
<td>CBP</td>
<td>Coexistence Beacon Protocol</td>
</tr>
<tr>
<td>FCC</td>
<td>Federal Communications Commission</td>
</tr>
<tr>
<td>SE</td>
<td>Stackelberg Equilibrium</td>
</tr>
<tr>
<td>NE</td>
<td>Nash Equilibrium</td>
</tr>
<tr>
<td>BER</td>
<td>Bit Error Rate</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Definition</td>
</tr>
<tr>
<td>--------------</td>
<td>------------</td>
</tr>
<tr>
<td>SNR</td>
<td>Signal to Noise Ratio</td>
</tr>
<tr>
<td>FHSS</td>
<td>Frequency Hopping Spread Spectrum</td>
</tr>
<tr>
<td>FEC</td>
<td>Forward Error Correction</td>
</tr>
<tr>
<td>RP</td>
<td>Received Power</td>
</tr>
<tr>
<td>SCH</td>
<td>Superframe Control Header</td>
</tr>
<tr>
<td>PC</td>
<td>Personal Computers</td>
</tr>
<tr>
<td>OS</td>
<td>Operating System</td>
</tr>
<tr>
<td>BAG</td>
<td>Bayesian Attack Graph</td>
</tr>
<tr>
<td>GPS</td>
<td>Global Positioning System</td>
</tr>
<tr>
<td>NOP</td>
<td>Non-Occupancy-Period</td>
</tr>
<tr>
<td>DoA</td>
<td>Direction-of-Arrival</td>
</tr>
<tr>
<td>EXP3</td>
<td>The Exponential-weight algorithm for Exploration and Exploitation</td>
</tr>
<tr>
<td>MAB</td>
<td>Multi-Armed Bandit</td>
</tr>
<tr>
<td>$S$</td>
<td>The set of all nodes on BAG</td>
</tr>
<tr>
<td>$S_i$</td>
<td>The $i^{th}$ node on BAG</td>
</tr>
<tr>
<td>$Pr(S_i)$</td>
<td>The probability of success of the attacker in reaching the $i^{th}$ node</td>
</tr>
<tr>
<td>$N_{ext}$</td>
<td>The set of external nodes on BAG</td>
</tr>
<tr>
<td>$N_{ter}$</td>
<td>The set of terminal nodes on BAG</td>
</tr>
<tr>
<td>$N_{int}$</td>
<td>The set of internal nodes on BAG</td>
</tr>
<tr>
<td>$\mathcal{E}$</td>
<td>The set of directed edges on BAG</td>
</tr>
<tr>
<td>$e_j$</td>
<td>The $j^{th}$ network vulnerability</td>
</tr>
<tr>
<td>$pa[S_i]$</td>
<td>The set of all parents of the $i^{th}$ node on BAG</td>
</tr>
<tr>
<td>$Pr(e)$</td>
<td>The probability of vulnerability exploitation</td>
</tr>
<tr>
<td>$Li_j$</td>
<td>The attack likelihood in exploiting the $j^{th}$ node</td>
</tr>
</tbody>
</table>
on BAG

$Im_j$ The attack expected impact when exploiting the $j^{th}$ node on BAG

$LCPD_i$ The local conditional probability distribution of the $i^{th}$ node on BAG

$\mathcal{R}$ The set of relations among parent nodes on BAG

$\mathcal{P}$ The set of discrete conditional probability distribution functions for every node $S_i \in N_{\text{int}} \cup N_{\text{ter}}$

$m_j$ Attack vector or attack strategy

$m_j$ Defense vector or deception strategy

$p^m$ The probability of success of attack vector $m$

$\mathcal{G}$ The Deception-Based Security Game

$A$ The Attacker

$D$ The Defender

$l_z$ The $z^{th}$ attack action in attack vector $m$

$Cl_z$ The implementation cost of attack $l_z$

$r_z$ The relative cost factor of attack $l_z$

$Cl$ The attacker’s cost unit

$L$ The maximum number of attack actions

$l_1$ The attacker launches the PUE attack

$l_2$ The attacker launches the masking attack

$l_3$ The attacker launches the blinding attack during the receiving times of the spectrum reports

$l_4$ The attacker launches the blinding attack during the receiving times of the spectrum decision

$N$ The maximum number of deception actions
A honeypot which protects the QP
A honeypot which protects the sensing reports
A honeypot which protects the spectrum decision
The implementation cost of honeypot $k_n$
The relative cost factor of honeypot $k_n$
The defender’s cost unit
The time required for sensing the spectrum
The time required for sending the sensing reports
The time required for sending the spectrum decision
Attacker’s expected gain
Defender’s expected gain
The probability of attack actions $l_z \in m_j$ falling into honeypots $k_n \in h_i$
The probability of the success of attack strategy $m_j$
Attacker’s return when capturing the channel
Defender’s return when capturing the attacker
Attacker’s expected loss due to falling into honeypots
The cost of relocating the identified attacker’s platform
Defender’s expected loss from attacked vulnerabilities
Attacker’s payoff function
Defender’s payoff function
The cost of implementing attack strategy $m_j$
The cost of implementing deception strategy $h_i$
The set of $A$’s best responses
The Attacker’s normalized payoff function
The Defender’s normalized payoff function
\begin{itemize}
  \item $I_A$ The Attacker's incentive factor
  \item $I_D$ The Defender's incentive factor
  \item $T_A$ The Attacker's incentive factor
  \item $\mathcal{M}$ The set of all attack strategies
  \item $\mathcal{H}$ The set of all deception strategies
  \item $\Sigma_A$ The attacker's mixed strategy profile
  \item $\sigma_{m_j}$ The probability assigned to attack strategy $m_j$
  \item $G_{\mathcal{R}}$ The repeated security problem
  \item $\epsilon$ The error upper bound
  \item $T$ The total number of repeated game rounds
  \item $r_i^{(B,t)}$ The defender's instantaneous algorithmic reward from playing deception strategy $h_i$
  \item $\mathcal{R}^{(B,t)}$ The defender's instantaneous reward from Algorithm $B$ at game round $t$
  \item $\mathcal{R}^B$ The defender's cumulative reward from Algorithm $B$
  \item $\mathcal{R}_{Mem}$ The defender's algorithmic reward history
  \item $\Psi^B$ The defender's worst-case regret when using algorithm $B$
  \item $\eta$ The learning parameter
\end{itemize}
Chapter 1

Introduction

1.1 Context and Background

The demand for the radio spectrum has grown explosively over the last decades due to the ubiquitous usage of wireless devices in accessing the vast range of new high data-rate consumer applications. In recent times, certain portions of the frequency spectrum have become remarkably overcrowded, especially in the Cellular band and the Industrial, Scientific and Medical (ISM) band. But substantive portions of the spectrum used for military, radars, public safety communications, and some commercial services, such as the Television (TV) bands, are widely underutilized [1].

Cognitive Radio (CR) is built on software-defined radio and can intelligently sense, manage, and access licensed spectrum bands which are temporarily not in use by the authorized licensees of the spectrum. In Cognitive Radio (CR) paradigm, PUs access the licensed spectrum any time they want without any concern of interference whereas, secondary users (SUs) can dynamically/opportunistically access the bands that are temporarily vacant, or not in use by PUs, without causing any violation to primary users’ communication capabilities [2,3]. Software defined radio (SDR) platform is used as a reconfigurable radio frequency (RF) front-end in the implementation of the CR physical layer where spectrum variations are sensed and transmitted to upper layers through the CR’s intelligent algorithms to control the opportunistic/dynamic spectrum access.

Nowadays, there exist multiple CR development frameworks which target the development of CR-based technologies to address the underutilized frequency bands. For instance, the United States military’s defense advanced research projects agency (DARPA) is developing the neXt Generation (XG) program. The XG program aims at developing wireless
systems which can dynamically redistribute allocated spectrum to improve military communications in severe jamming conditions [4]. The IEEE 802.22 standard is another example of a commercial CR-based network\(^1\) (CRN) that utilizes cooperation in sensing the spectrum. The IEEE 802.22 was issued by the IEEE working group on wireless regional access network (WRAN) to address the opportunistic use of the spectrum in TV and wireless microphone bands [2]. It utilizes the point-to-multipoint architecture with a central entity called the **cognitive base station** (CBS) and several peripheral nodes called the **customer premises equipment**\(^2\) (CPE). Figure 1.1 shows the management reference architectural model of the IEEE 802.22 CR networks with coexisted neighbored CRNs. The CBS controls the opportunistic spectrum access of the CPEs within its cell. Moreover, in cases when the available

---

\(^1\)The term CRN means the network established by SUs only, and it does not include any communication with the primary user other than authenticating the PU user’s signal.

\(^2\)We will refer to the *customer premises equipment* (CPE) as the secondary users (SUs), henceforth.
Figure 1.2: Cognitive radio functionality under Cognitive Cycle (CC)

channels are less than the required channels by CBSs, the self-coexistence mechanism can be used to establish collaboration between CBSs with overlapped coverage areas through channel time sharing. In this case, the neighboring CBSs are allocated to non-interfered subsets of frames in the super-frame, which lowers the overall throughput [5].

The SUs coordinate, in general, their actions by negotiating on the available frequency channels and the network quiet periods (QPs). In QPs, the spectrum sensing process takes place, and no transmission from SUs is allowed. Such a coordination is realized by connecting media access control (MAC) layers from different SUs through a communication channel known as the common control channel (CCC). The SUs submit their spectrum sensing reports to a central entity, named as the cognitive base station (CBS) where the spectrum decision is fused, then sent to the cooperating SUs. The sensing cycle is a real-time process that involves spectrum sensing and spectrum negotiation among CRN’s entities before valuable communication takes place [2] as shown in Figure 1.2. In the first phase (sensing), the spectrum is widely sensed for the presence of primary users or other secondary users. In the second phase (analysis), the detected environment information is processed and charac-
terized. In the third step (reasoning), the processed information is utilized in making the decision on whether or not to use the spectrum at specific times and locations. In the last phase (adaptation), the radio parameters are reconfigured to achieve reliable communication for the secondary users’ network.

1.2 Cognitive Radio Security Challenges and Opportunities

CR security is the study/assurance of CR functionality under presence of malicious (misbehaving) users. CR solution entails new security challenges, as well as existing conventional security concerns that CR shares with other wireless networks [6]. Accordingly, the security threats that CRNs are vulnerable to come in two types:

i) traditional threats like naive jamming and eavesdropping that exist due to wireless channel and that affect mainly the physical and the MAC layers, and

ii) CR-specific threats that exist due to CR’s unique characteristics such as spectrum sensing, hardware reconfigurability, spectrum rules learnability, and the usage of a common channel for side communication among the SUs [6–11].

Nevertheless, CRN’s security vulnerabilities (i.e., weak points) and threats (i.e., attacks) continue to increase due to the rapid growth in attackers’ capabilities. The increase in CRNs’ security vulnerabilities renders the need to conduct more research in this area such that the CRN’s main security vulnerabilities from probably misbehaving users are dealt with, thus helping to ensure the future availability of CR solution [12].

In relation to the CR network security domain, one primary challenge is to efficiently represent possible multiple security threats, and assess their combined effects. Most of the existing research efforts in the area of security threat assessment of CR systems solely examine the issues of denial-of-service (DoS) attacks, such as primary user emulation (PUE) attack and control channel jamming attack, each treated in isolation. The PUE attack takes place when one or more secondary users mimic the presence of a primary user by emulating its signal characteristics. This attack disrupts the CRN operation and forces the
CRN to vacate the frequency band (handoff) [13]. Attackers launch PUE attack either to
gain exclusive access to some parts of the spectrum or to cause harm to the CRN. Generally
speaking, PUE attack causes degradation to the spectrum utilization but if the attacker is
granted exact information about the network’s QPs and the free channels list, through e.g.
eavesdropping on the Common Control Channel (CCC), a complete Denial of Service to
CRN can be easily caused.

In CR paradigm, the participation of the PU makes the problem of detecting and mit-
igating the attacks mentioned above rather hard. Moreover, the Federal Communications
Commission (FCC)’s regulation prohibits any modifications to PU’s signal which makes the
detection process of such attacks more challenging [14].

Radio jammers deliberately transmit radio jamming signals to:

1. Impair the spectrum sensing process, named as intelligent jamming, or

2. Cause a degradation in the quality of the received data during useful commu-
nication time, referred to as naive jamming.

The intelligent (acute) jammers continuously perceive the targeted frequency channel(s)
(expressed only as the channel(s) henceforth) to gain exact information about the CRN’s
sensing cycle and only attack at CRNs’ vulnerable times to cause maximum damage while
saving power and decreasing the probability of being detected. Previous studies proved that
acute jammers with full information on the schedule of QPs and the free channel lists could
easily cause a complete denial of service (DoS) to CRNs [14].

An acute radio jammer would target impairing the victim’s CR receiving-circuitry or
cause actions to circumvent the victim’s spectrum CR sensing circuitry [11,13]. The victim’s
receiving-circuitry is impaired by transmitting sufficiently high power continuous white noise,
to decrease the signal-to-noise ratio (SNR) during i) the receiving times of the spectrum
sensing reports, to disrupt the cooperation in spectrum sensing, and ii) the receiving times
of the spectrum decision, leading to the isolation of the SUs who are within the jammer’s range.

A radio jammer’s actions that may cause the circumvention of the victim’s sensing circuitry include: emulating the PU’s signal during the QPs when the PU is not using the channel, forcing the CRN to vacate the channel (handoff). A radio jammer can also mask the PU signal by transmitting sufficiently high power continuous white noise during the QPs while the PU is using the channel. Masking PU’s signal induces the CRN to access the channel, with a consequence of possible penalties on the CRN due to the violation of the rights of the PU.

In essence, it is feasible for one attacker equipped with state-of-the-art CR platforms to launch various combinations of the jamming attacks mentioned above by a slight change in the malicious configurable radio. Moreover, a group of jamming attackers can coordinate their actions to increase the expected impact on the CRN, which alternatively, can be considered as one attacker who can launch multiple malicious threats simultaneously. The problem of a sophisticated acute jammer who can launch different combinations of jamming attacks against CRN’s sensing and receiving circuitries in the same CRN’s sensing cycle is left unaddressed in the literature.

The last challenge we consider in this thesis is related to the frequent interactions between the defending CRN and the attacker(s), and the robustness of the calculated game solution. In practice, the jamming attacker(s) might attack the victim CRN multiple times per second. In addition, the attacker(s) might chase the CRN over the spectrum to re-engage\(^3\). Thus, the interaction with the advanced jamming attackers takes place frequently at repeated intervals [15, 16]. The repeated security game over the spectrum can be viewed as an opportunity to compensate for the *subjective* errors which might exist in the calculated game solution.

\(^3\)A.k.a. the frequency-follower jammer and is designed to target the frequency-hopping-based networks.
1.3 Problem Statement and Thesis Objectives

By and large, the main unaddressed research problems in the area of CRNs’ security include:

i) the finding of a reasonable security metric for CR networks which can assess the CRNs’ security vulnerabilities and threats under the assumption of the coordinated DoS attacks.

ii) The mitigation of the coordinated acute jamming attacks wherein different types of acute jamming attacks collude to increase their negative impact on the victim CRN. Finally, iii) the consideration of the case when the defending CRN is learning the optimal defense response under the assumption of a series of successive coordinated acute jamming attacks.

In this thesis, the important and non-trivial problem of assessing and mitigating the multiple coordinated jamming attacks in cognitive radio networks is investigated and solved. The attack mentioned above is critical because it is preferable for the attackers to coordinate their actions to maximize the negative impact on the victim cognitive radio network. The main challenges faced in this thesis include the finding of a quantifiable metric that can assess the security of cognitive radio networks, which helps in guesstimating the most viable security vulnerabilities. Another challenge is the design of a security mechanism which can proactively decrease the security threat level in cognitive radio networks by protecting main network’s security vulnerabilities.

To sum up, for the first time in the literature of CRNs’ security, the fundamental goal of this thesis is to reduce the combined effect of contingent acute jamming attacks in cognitive radio networks. To achieve the thesis’s goal and, without any loss of generality because the IEEE 802.22 is used as a basis, particular thesis objectives are as follows.

First, performing an up to date security threat assessment of the IEEE 802.22-based CRNs forms the focus of Chapter 3 wherein key known CRN’s vulnerabilities are analyzed, the authors of [17] proved that sum of DoS attackers’ payoff could be enlarged by approximately 10–15% if they coordinate their actions. In addition, it is shown in [18] that the DoS probability in the IEEE 802.22 cognitive radio networks increases by 51.3% when the DoS attackers coordinate their actions.
and both likelihood and severity of the aftermaths of probable CRN’s security threats are formulated as an attack model\(^5\).

Second, another primary challenge is the acute radio jamming attacks as one of the most effective CRNs’ security threats as indicated earlier. The mitigation of the contingent acute radio jamming attacks (deceiving attack) forms the focus of Chapter 4. In particular, a Stackelberg game that utilizes deception against the deceiving attack is proposed and a game solution (i.e., a specific deployment probability of defense actions) that guarantees a certain payoff for the defender and the attacker under game equilibria is calculated.

Thirdly, in the preceding research objective, the calculated game solution mainly suffers from \(i\) the sensitivity to the errors in the assumed attacker’s behavioral model, and \(ii\) the inflexibility to the change in the estimated attacker’s behavior when repeating the game for a period of time. Overcoming these shortcomings in a repeated game framework is the focus of Chapter 5.

1.4 Contributions and Outline

It is evident from the preceding that the problem of contingent acute jamming attacks is significant because its solution directly results in a reduction in the probability of denial of service to the CRN. The contributions of this thesis concerning the research problems stated in Section 1.3 are listed as follows:

1. In assessing the security of CRNs presented in Chapter 3, the main contributions are threefold. First, by introducing the Bayesian attack graph (BAG) model as a reasonable security metric for CR networks, we create both the attack graph (AG) and BAG model representations of the DoS attacks in the

\(^5\)The attack model is a possible way of abstracting the security threats through representing threats scenarios, their predictable consequences, and the likelihood of occurrence by using graphs, trees or block diagrams.
context of the IEEE 802.22 networks. Second, we compute the DoS probability of simultaneous multiple attack scenarios and the probability of exploiting known vulnerabilities of the IEEE 802.22. Third, we pinpoint the most probable DoS attack path of the IEEE 802.22 networks. To our best knowledge, this is the first work that introduces the BAG model as a single and sufficient quantitative metric to assess the effect of simultaneous multiple DoS attacks in CR networks. It is worth noting that the proposed AG and BAG model representations are not etched in stone and can be further enhanced in the future through the addition of new threats and newly discovered CR networks’ vulnerabilities which contributes to the importance of the utilization of such a security threat representation model.

2. The main contributions in Chapter 4 regarding the mitigation of contingent acute jamming attacks are 
   i) the introduction of the deception-based defense strategies which could decrease the probability of success of the deceiving attacks to nearly 0% when the defender has a high incentive to protect the channel.  
   ii) The presentation of the derivation of the closed-form expression for the Stackelberg equilibrium (SE) when the PU activity pattern is a common knowledge in the game.

3. Finally, with the assumption of a repeated game play, the main contribution of Chapter 5 is the introduction of six hybrid algorithms which combine both the advantages of the game theoretic solutions and the online learning algorithms. The proposed algorithms enjoy an efficient theoretical regret upper bound and a very good initial behavior with respect to celebrated standard online learning algorithms in both cases of the defender’s feedback structures. The proposed algorithms achieved up to 92% decrease in the initial per-round regret in comparison to the standard learning algorithms.
The remainder of this thesis consists of five chapters, outlined as follows. Chapter 2 reviews the previous work pertaining to the research problems presented in this thesis. In Chapter 3, the security threat assessment of the CRNs using the BAG model is calculated and major DoS attacks are pinpointed.

In Chapter 4, the proposed deception-based Stackelberg game-theoretic defense scheme is introduced and evaluated under the assumption of the contingent acute jamming attacks. In Chapter 5, the defense mechanism mentioned-above is integrated with the online learning to reproduce a robust solution in a repeated game framework. Finally, the thesis is concluded in Chapter 6 which presents the research principal findings, potential limitations, and directions for future works.
Chapter 2

Related Work

Cognitive radio (CR) security is the study/assurance of the CR functionality under the presence of malicious (misbehaving) users. The infancy of the CR technology calls for rigorous investigation on probable security vulnerabilities and corresponding mitigation techniques [6–11].

Most contributions targeting cognitive radio network’s (CRN) security issues have focused on combating certain types of attacks such as primary user emulation (PUE) attack, control channel saturation, and eavesdropping attack [6–11,19,20]. However, sophisticated attackers can simultaneously use a combination of the denial-of-service (DoS) attacks. Also, attackers can initially target one CRN’s vulnerability before switching to target another one, which can be described as attack path. New threats arise more frequently, with the fast growth of attackers’ capabilities as well as of CR applications [20], creating a strain on CRN’s security system to take measures accordingly.

In addition, the game theory was used in many works in the literature as a mathematical tool in understanding and modeling the security problems, thus helping the security engineers to tighten the security plans pragmatically [16, 21–27]. Broadly, a security game problem is a mathematical formulation of the possible interactions between the defender(s) and the attacker(s). The game solution is a deliberated description of the possible game outcomes for each player in the game [28].

This chapter presents a review of the existing works of relevance to cast the proposed work in the context of state of the art. The proposed work in this thesis has a harmonious relationship with three lines of research in the literature. The first line of work focuses on assessing the combined effect of the multiple coordinated Denial of Service (DoS) attacks.
The second line of research looks into providing game theoretic solutions for the security problem in the context of sophisticated attacks in CRNs. Finally, the third line of research discusses the case when the players interact with a high frequency in repeated plays.

2.1 Security Threat Assessment of the IEEE 802.22 CRN

At present, only few research efforts targeted CR network security threat assessment. In a series of works [29–31], authors logically represented potential CR network DoS attacks and vulnerabilities using the Hammer model [32]. The results introduced in the aforementioned works on the CRNs’ security assessment formed the basis of the security structure of the IEEE 802.22 standard. In [5] and [33], the IEEE 802.22 security ad–hoc group assessed CR networks’ functions and algorithms for potential vulnerabilities. Security features were proposed to countermeasure possible adversary breaches of the IEEE 802.22 standard.

In [20], an approach to mitigate the security threats in CRN was discussed. The approach analyzed CRN’s vulnerabilities and proposed certain mitigation techniques for some example threats. In [34], authors instituted a cognitive cycle–based environmental threat management engine. Targeting the development of the operating software defined radio (SDR)–based Positive Train Control (PTC) system. Moreover, the same authors in [35] outlined a possible leveraging of their past work to accommodate for adversarial activities through tracking the thresholds of selected radio parameters, such as bit error rate (BER), signal to noise ratio (SNR) and received power (PR). In [17], the behavior of coordinated DoS attackers in the IEEE 802.22 networks was studied, where sophisticated attacker or multiple attackers can sequentially/concurrently target different CR vulnerabilities to disrupt CR network communication. The authors used the cooperative game theoretic approach to formulate the problem. Eventually, they demonstrated that the sum of all attackers’ gain could be enhanced by approximately 10–15% if they chose to coordinate their malicious

6PTC is a distributed communication and control system for USA’s railways
Also, the authors of [30] presented a general view of CRNs’ security threats with respect to the targeted layer in the communication stack. In addition, potential CRNs’ security vulnerabilities are pointed out and discussed from the perspective of confidentiality, integrity, and availability (CIA triad) security model\textsuperscript{7} during the times of CRNs’ useful communication. Moreover, advances on CRNs’ security threats and countermeasures can be found in [6, 7, 10, 36, 37], and [38]. In addition, the identification and the analysis of CRNs’ security vulnerabilities can be found in [20, 29, 35] and [39].

None of those mentioned earlier works considered the effect of simultaneous multiple attack scenarios from the CRNs’ perspective, in spite of the high importance of such an approach in guiding CR systems designers to build reasonable security tightening plans with optimum countermeasures.

In a different context, the attack graph (AG) is a security model that abstracts the cause–consequence relationship among potential threats, known network vulnerabilities, and attackers’ goal(s) [40]. The AG model is extensively adopted by the security engineers of Cyber Networks in analyzing network security [41]. A compact probabilistic model representation of AGs is the Bayesian Attack Graph (BAG), based on the Bayesian notion which captures the attack’s likelihood of exploiting network vulnerabilities and probable attack paths [42].

In this part of the thesis, we create the BAG model representation of the adversarial–based DoS attacks in the IEEE 802.22 networks that utilize TV bands. The generated BAG model representation can be used to compute the probable contingent security threats from the complex domain of all possible threats according to the behavior of a rational attacker. However, this work only considers the adversarial–based DoS attacks, benign threats,\textsuperscript{7}\textsuperscript{7}CIA model is a an important security model which summarizes the system status with respect to critical security requirements. In particular, data confidentiality designed to guide policies for information security within an organization.
such as noise, interference, and hardware failures can be considered by CR networks’ security planners through increasing the computed expected DoS probability due to malicious actions by a deterministic percentage.

2.2 Deceiving the Deceivers in CRNs

Recently, the detection/mitigation of the acute jamming attacks in CRNs has attracted the attention of wireless security researchers. Examples of the main approaches to overcome the impairment of CR’s sensing circuitry are the following.

First is the clustering–based approach, such as the work in [13], where individual SUs report the existence of PU’s signals to the CBS. Then, the CBS takes the spectrum decision by giving different weights to SUs’ reports according to their relative locations and some trust factor in order to maximize the legitimate PU detection probability. Also in [43], each SU iteratively exchanges/updates its belief about a particular activity on the channel, being an adversary or not, with neighboring SUs. According to the authors in [43], after a sufficient number of observations, convergence to a final belief is guaranteed. The main problem with this approach is that it may be costly from the point of view of the required number of observations or communication overhead to detect an attacker.

A second method for overcoming the CR’s sensing circuitry impairment is the game theory based approach. In [44] and [45], the authors utilized game theory to deduce a closed-form of the Nash equilibrium between CRN defender who uses surveillance-based defense strategies and PUE attacker. The results showed the strong influence of the players’ gain-to-cost ratio and the availability of the channel on the game equilibrium. Also in [46], the interaction between PUE attacker and SUs was modeled using game theory and the optimal SU’s sensing strategy that maximizes the channel usability was obtained.

In a different context, many works addressed the detection and mitigation of the acute radio jamming attacks that target the CR’s receiving-circuitry. For instance, in [47], the
impact of jamming attacks on the cooperative spectrum sensing process was investigated. And an anti-jamming technique was proposed that utilizes a hybrid forward error correction (FEC) code. In [48], a Stackelberg-based game theoretic approach was used to model sophisticated jamming/anti-jamming scenarios between CRN and radio jammers that are capable of sensing multiple channels simultaneously (robust spectrum sensing capability, as called by the authors). The frequency hopping spread spectrum (FHSS) was proposed to mitigate the impact of jamming attacks. In addition, in [49], a zero-sum stochastic anti-jamming game problem for CRNs was formulated. The authors utilized the channel hopping as the defense scheme and the minimax-Q as the learning algorithm. The results showed a better overall spectrum efficient channel throughout.

Broadly, deception techniques such as honeypots were widely exploited in the area of Cyber Networks security to facilitate the understanding of malicious actions and behavior of attackers [50–54]. However, the well-informed attacker is assumed to know about the honeypot types and numbers in the network, but still the exploitation of honeypots decreases the attack’s likelihood through increasing the attacker’s uncertainty about the system vulnerabilities [55]. In the context of CRN, the authors in [56] proposed a dynamic assignment mechanism for honeynode secondary users to deceive the jamming attackers. The sacrificed secondary user, honeynode, acts like a typical data transmitter to attract the attacker to this channel, accordingly, obtain the attacker’s fingerprint.

Finally, one well-known anti-jamming approach applies the spatial filtering with beam-forming antenna arrays technique [57]. In this method, the direction of arrival of jamming signals is detected based upon the intrinsic differences between legitimate signals and jamming signals. The jammer’s direction is then used to modify the antenna array’s pattern, placing the jammer in the nulls of the antenna [58] and [59].

The importance of the methods mentioned in the preceding lies in modeling and mitigating the jamming attacks, but none of them considered the case of sophisticated attackers
who can simultaneously launch a combination of different types of jamming attacks (contingent jamming attacks) targeting CR’s receiving and sensing circuitries in a CR sensing cycle. Contingent acute jamming attacks form the motivation for this part of the thesis.

The impact of contingent jamming attacks on CRNs is much higher than the most severe sole jamming attack as was proven by the authors in [18] and [17]. Particularly, in [18], the probability of DoS to the IEEE 802.22 network under the assumption of contingent attacks, from the CRN perspective, was estimated to increase up to 51.3% to the most severe sole DoS attack. While in [17], the coordinated DoS attacks problem on the IEEE 802.22 networks were investigated from the attacker’s perspective. Therefore, the authors showed that the attackers could attain as high as 10-25% more net payoff when coordinating their actions in comparison to the case when they do not cooperate. In addition, in Chap 3, the security assessment process indicates up to 43% increase in the probability of DoS in CRNs considering contingent jamming attacks in comparison to the most severe sole jamming attack.

With the growth of attackers’ capabilities, it is feasible for one attacker equipped with state-of-the-art CR platforms to launch various jamming attacks by a slight change in the malicious configurable radio. Moreover, a group of jamming attackers can coordinate their actions to increase the expected impact on the CRN, which alternatively, can be considered as one attacker who can launch multiple malicious threats simultaneously. The contingent acute jamming attacks are referred to as the deceiving attack henceforth, and the mitigation of such an attack forms the focus of Chapter 4 of the thesis work. The proposed deception actions (honeypots) proactively and collectively decrease the likelihood of the deceiving attack in CRNs.

This work differs from work in [56] in that the deception is utilized to protect both of CR’s sensing circuitry and receiving-circuitry. Besides, this work is different from the works in [48] and [45] mainly in consideration of contingent jamming attacks. To the author’s
best knowledge, this is the first work that utilizes deception in protecting CRN’s security vulnerabilities. It is shown in this part of the thesis that a defender with high incentive to defend the channel can reduce the probability of success of the deceiving attack to nearly 0% by using the proposed defense mechanism.

Using the IEEE 802.22 CRN as a basis, a Stackelberg-based game theoretical framework that utilizes deception against the deceiving attack is proposed. One might wonder why using Stackelberg model over Nash model in this work, and the reason is twofold [60] and [61]. First, in the Nash model, players are assumed to calculate (or expect) the equilibrium before actually playing the game, and then play the calculated equilibrium. Strictly speaking, the attacker is needed to be aware of the defender’s payoff function and the actions space to calculate the equilibrium, which is impractical in many situations. Second, the Nash model assumes no bias in information among players, meaning, the attacker cannot observe the defender’s actions before playing the game, which is much hard to justify in practice. These two reasons make the Stackelberg model the core of many implemented real-world applications, e.g., [62] and [63].

Notice that one of the motivations of the proposed work is the application of the Stackelberg game model in real-world security domains. The work in [62] presents the ARMOR security system where the Stackelberg model is used for deploying police checkpoints along the roads connecting the gates of Los Angeles International Airport (LAX). Another application is the IRIS security system [63], where the Federal Air Marshals Service (FAMS) utilizes the Stackelberg model in assigning the armed officers to the commercial air flights to counteract terrorist attacks. In the literature of wireless networks, the Stackelberg model was utilized in modeling the interactions between the jamming attacks and the wireless networks, such as the works in [64] and [54].

It is most important to note that while the players in the proposed Stackelberg game interact simultaneously with each other, the attacker does not know the exact schedule of
the defense actions (honeypots) but has only knowledge of the probability distribution over the honeypots by observing the defender’s actions in hindsight.

It is quite common in the literature of security games to formulate players’ utility functions as a minimization of the total-loss in a zero-sum framework [45]. The reason is to represent the strictly competing nature of the players, yet assuming each player’s gain/loss is equitable to other player’s gain/loss. Another reason for such a formulation is to represent the tendency of the attacker and the defender to minimize the defense and attack budgets, respectively. Nevertheless, in this thesis, players’ objectives are still conflicting, but players’ gains/costs are not assumed to be precisely balanced, thus forming a general-sum game instead of a zero-sum game. Furthermore, the players’ utility functions are formulated as a maximization of payoff functions where the defender gains more from capturing the attacker and the attacker gains more by avoiding the deployed honeypots. This formulation is more realistic as it provides the flexibility to consider independent defense and attack incentives in different game scenarios.

2.3 Learning in Repeated Games

Mainly due to the dynamicity of the security threat environment and the embedded subjective assumption about the attacker’s model and behavior, the calculated security game solution (the game equilibrium) is acknowledged to be inaccurate and just an approximation of the exact game solution. The issue above was exposed in multiple works in the literature, for instance in [10, 23, 65–69]. Main approaches which address the aforementioned challenge include:

First, the model-based approach where the defender’s incomplete information (which forms the defender’s belief about the existence of attackers with diverse models or behaviors) is used as a fixed probability distribution over the attacker’s types. The Bayesian game is then formulated by converting the game of incomplete information into a game with complete
information using the Bayes-rule. The utilization of the Bayes-rule targets finding a better game solution which is robust to changes in attacker’s behavior from the expected attacker’s model. Mostly, the defender might gain the prior probability distribution over attacker’s types through learning.

Examples of works that utilize the approach mentioned above include [23, 44] and [70]. In [70] the authors investigate the case when it is unknown to the defender what game model the attacker intends to play (i.e., Nash or Stackelberg model). The work in [70] assumes that the attacker can play:  

i) the Nash model with a probability \( q_1 \),  
ii) the Stackelberg model with a probability \( q_2 \), and  
iii) the No-attack strategy with a probability \( q_0 = 1 - q_1 - q_2 \).

The authors of [70] applied the Bayesian approach to find the equilibrium strategies under the above assumptions when the game is repeatedly played.

The work in [23] targets increasing the defender’s ability to pigeonhole the attacker’s type in order to estimate the attacker’s next step in the next game round. The game starts with a-priori probability distribution over a set of attacker types\(^8\). Then, the round by round revealed information about the attacker’s preferences is exploited in updating the defender’s belief on the attacker types.

In [44], a Bayesian game was introduced to model uncertainty about the attacker’s type, being a selfish attacker with probability \( \delta \) and a malicious attacker with probability \( 1 - \delta \). The Harsanyi model was utilized to address such an uncertainty where a third player is added to the game (named as nature) and she, nature, who decides the attacker’s type at the beginning of the game [71]. The game introduced in [44] is a single-run game.

Another type of uncertainty that might exist in the security games of CRNs is the uncertainty about the PU’s activities over the frequency channel. In Chap 4, the Harsanyi model is used in addressing the uncertainty on PU’s activities over the channel being busy (PU’s is using the channel) with probability \( \zeta \) and vacant (PU’s is not using the channel) with

---

\(^8\)The attacker types here is a general expression which refers to malicious opponents with different responses that might be from attackers with different constraints or capabilities.
probability \((1 - \zeta)\). Thus, the game solution was calculated in a single-run game scenario.

In addition to the types mentioned above of uncertainties, the Stackelberg-based game theoretic model assumes the attacker can observe the defender’s strategy correctly, which is not entirely practical. Thus, there have been lots of recent works in the literature that investigate the assumed attacker’s perfect observation, done mainly through introducing the Bayesian games, such as the works in [21,22,72].

There exist several works in the literature of security games that investigated the problem of robustness in the game solution in a different way. In [73] the authors propose a unified computational framework for handling various types of uncertainties in Stackelberg security games. Authors of [73] discussed three key uncertainty types: 1) the uncertainty about attacker’s payoff, 2) the uncertainty about defender’s strategy, and finally 3) the uncertainty about attacker’s rationality. Based on the introduced unified computational framework, authors introduced a set of robust algorithms which address different combinations of the above mentioned uncertainties. The numerical results of the proposed algorithms in [73] proved an enhancement in the performance and the quality of the Stackelberg game solutions.

Also, the work in [74] introduces an interval based approach to model uncertainty in large security games. In [74], the attacker’s payoff is assumed to lie within a known uncertainty interval. Thus, the maximum regret for the defender under the worst case condition is used to find the defender’s optimal strategy.

Broadly, the model-based approach mainly suffers from the following:

1. The tremendous increase in the size of the game problem when the number of underlying attacker’s behavioral models are increased.

2. The lack of resilience to the dynamic threat environment when the attacker’s incentive, thus behavior, is changing over time.

The second approach is the model-free approach where no prior assumptions on the attacker’s model or behavior are made before playing the game, such as the works in [75–77].
In [75] and [76], authors introduced learning algorithms which are based on the celebrated Follow-the-Perturbed-Leader (FPL) prediction algorithm [78] in a bandit feedback environment when the defender is suffering losses and collecting rewards, respectively. The numerical results of [75] and [76] proved an efficient conversion against the optimal adaptive defense strategy in hindsight. Also, the algorithms mentioned above enjoy theoretical performance guarantees and asymptotically tends to zero regrets when the game runs indefinitely. The work in [77] introduces two online-learning-based algorithms that apply to the full feedback structure (where the defender observes the attacker who responds), and the bandit feedback structure (where the defender observes the attacked target only).

Fundamentally, the model-free approach despite being a suitable approach to address the dynamicity of the attacker’s behavior, yet it suffers from the following:

1. A weak initial performance.

2. The inflexibility in considering a-priori information which might be available from the security experts or from other imperfect solutions which might be introduced by game-theoretic based security algorithms.

In a different context, the proposed work in this part of the thesis is closely related to two primary online learning algorithms from the area of machine learning. First, the proposed work is close to the standard learning algorithm with full information, named the HEDGE algorithm [79]. Remarkably, the authors of [79] proved that performance of the HEDGE algorithm is almost as good as the best defense strategy in hindsight.

Second, in the partial (bandit) feedback settings, the proposed work is closely related to the Exponential-weight algorithm for Exploration and Exploitation (EXP3) [80]. Importantly, the EXP3 algorithm is a variant of the HEDGE algorithm, in particular, an important sampling step is added in the EXP3 algorithm which helps in estimating the unreceived feedback information. The EXP3 algorithm is considered the most pessimistic online learning algorithm due to its weak assumptions about the defender’s feedback structure [81].
With the target to overcome the shortcomings in the above-described model-based and model-free approaches, we propose a set of hybrid algorithms in which the online learning is integrated with the game solution (game equilibrium) in a repeated security game framework. The proposed hybrid-algorithms enjoy the following:

1. An efficient theoretically-guaranteed regret upper bound in comparison to the best fixed pure strategy in hindsight.

2. The resilience to the changes in attacker’s behavior during repeated game runtime.

3. A good initial response in comparison to the pure online learning algorithms.

The closest work in the literature to the proposed work is in [69] from the area of Border security. The authors of [69] introduced a set of learning algorithms which combine the game equilibrium with the online learning in a game that models different border patrolling scenarios.

The differences between the proposed work and the work in [69] mainly include: i) the regret upper bound of the proposed work is theoretically guaranteed. ii) The game theoretic solution is partially considered in the proposed hybrid algorithms as an expert opinion in an online learning framework. iii) The game setup in the security of CRNs requires a faster response from the defender as the game might span over multiple seconds representing nearly 60 interactions among game players. However, in [69], the game might span over multiple years with thousands of interactions with the border smugglers.
Chapter 3

Security Threat Assessment of The
IEEE 802.22 CRNs

3.1 Introduction

This chapter highlights potential CRN’s security threats and vulnerabilities and assesses the probability of success of multiple DoS attacks when simultaneously launched against the IEEE 802.22 CRNs. The main contribution of this chapter is the introduction of the Bayesian Attack Graph (BAG) model as a reasonable security metric for CR networks. Moreover, the results confirmed the importance of protecting the cooperative spectrum sensing process being a prime target for the attackers in the IEEE 802.22 CRNs.

As discussed in Chapter 1, the IEEE 802.22 is the first complete standard for wireless networks that utilize CR technology [4, 82]. Two security sub-layers are adopted in the IEEE 802.22 standard as follows:  

i) the security sub-layer1 that maintains data confidentiality, integrity and authenticity; and  

ii) the security sub-layer2 which is mainly for protecting the rights of incumbent users in conducting interference-free communication.

Notably, the incumbent users are defined in the IEEE 802.22 standard as  

i) the primary TV transmitters,  

ii) other cognitive base stations which occupy the channel,  

iii) the wireless microphones and  

iv) the IEEE 802.22.1 cognitive beacons [3]. Primary incumbent-protection mechanisms of the IEEE 802.22 standard include  

i) the signal classification and detection techniques,  

ii) collaborative spectrum sensing,  

iii) incumbent database corre-

---

lation with geo–location information and *iv*) spectrum decision making [3,5]. Define the DoS attacks in IEEE 802.22 as the *adversarial* actions that may partially/completely prevent CR networks’ communication over targeted frequency channel/band, or time slot [29]. Potential DoS attacks of IEEE 802.22 networks in the TV bands are described in the following subsections.

3.1.1 Impairment of Sensory Information

One of the most important mechanisms in protecting incumbent users is through enabling both on–board sensing techniques and intra–cell distributed sensing mechanisms. Consequently, impairment of sensory information is a *prime* target for multiple attackers, such as *primary user emulation* (PUE) attack and *coexistence beacon protocol* (CBP) packets falsification attack [7, 83, 84]. The impairment of sensory information has a *High* impact on the entire CR network. Improving the sensory abilities, in the context of discriminating between legitimate and malicious users, can be achieved by considering: *i)* more accurate signal classification and detection techniques, e.g. legitimate transmitter’s hardware fingerprint–based approaches [85] and *ii)* enhanced collaboration strategies that can effectively detect and isolate outliers [19, 86]. Moreover, breaching the self–coexistence protection mechanism by attackers is another form of DoS. As it targets preventing CR network from exploiting certain *portions* of the available frequency channels during specific time slots. Notably, for an attack to successfully impersonate an overlapped IEEE 802.22 cell, it *must* spoof the spectrum sensory information, through transmitting valid superframe control header (SCH) or CBP packets and distort the signature authentication mechanism which is a *sophisticated* process [3,83,87].

3.1.2 Location Falsification and Location Failure

The second line of defense in protecting the incumbent’s rights in conducting interference—free communication is through correlating the CPEs locations with the incumbent
database. Intrinsically, CPE(s) within the protected contour of legitimate users is (are) prohibited from exploiting the incumbent channel and the first adjacent channels [3]. If the attacker succeeded in spoofing the CPE’s location such that the spoofed position falls within the protected contour of a PU, then the CBS will prohibit the victim CPE(s) access to the channel. In addition, if the CPE’s location information is inaccurate or unavailable, again, the CBS has to prohibit the CPE’s communication till it retrieves the location service [29]. Typically, the CBS’s location is fixed in the IEEE 802.22, which limits the problem to individual CPEs. Expediently, jamming GPS signals is an attack that can target such vulnerability, affecting all CPEs within its jamming range [88].

3.1.3 Control Channel Failure

In IEEE 802.22 paradigm, data and control frames are exchanged among network entities using the same frequency channel (split phase control channel).Attackers can selectively target the frequency channel during i) the transmission of the spectrum sensing reports by CPEs and ii) the transmission of the spectrum decision by CBS [39]. Again, jamming is the most likely attack that can exploit these vulnerable points in time, causing either high or limited impact on the network when its victim is the CBS or the CPE(s), respectively.

3.1.4 Databases Failure

According to the IEEE 802.22 standard, the CBS must have access to policy and incumbent database services. Typically, the incumbent database must be regularly updated (default update timer value is 24 hours), through certified incumbent database providers over the Internet. If the CBS’s stored incumbent database is obsolete and it was unable to access the incumbent database for a specified time (default value is 1 hour), the CBS shall de-register its associated CPEs and terminate the entire cell communication until it retrieves the database service [3]. Broadly, an attacker may exploit this vulnerable property and deny the CBS’s connection to database service, forcing a DoS to the entire network. However, it is only
effective when the on–board stored database is inaccessible or obsolete [31].

3.1.5 Deception of Databases

An intelligent attacker who can access the CBS’s stored incumbent databases may falsify the incumbent protection contours or the channels’ availability information. Consequently, the CBS/CPE(s) communication over specific frequency/band is prohibited until the database is re–updated.

3.1.6 Spurious Operating System Commands

Typically, the CPEs are as vulnerable as the personal computers (PC) to operating system (OS) viruses, malware and OS disconnects, which can root unpredictable actions from infected network entities, including wrong configuration of the victim’s air interface that deny communication over target frequency/band [20,87].

3.2 The Bayesian Attack Graph (BAG) Model

The Bayesian Attack Graph (BAG) is a model representation of a systematic method that incorporates visualizing, analyzing and quantifying probable security threat(s) that can target known network vulnerabilities to achieve particular adversarial goal(s). The Bayesian notion captures the dependencies among attack(s), potentially exploited network vulnerabilities and attackers’ goal(s). In the sequel, the BAG model components are introduced and explicitly discussed. Then a simplified BAG, representing two possible attacks of CR networks is presented and analyzed to demonstrate the concept.

3.2.1 BAG Model’s Main Components

Typically, the BAG model is formed by four main sets: \( S, E, R \) and \( P \), where, first, \( S \) is the set of BAG nodes, such that:

\[
S = N_{\text{ext}} \cup N_{\text{int}} \cup N_{\text{ter}} \tag{3.1}
\]
Table 3.1: Likelihood Component Evaluation Grid [5,31]

<table>
<thead>
<tr>
<th>Li&lt;sub&gt;j&lt;/sub&gt;</th>
<th>Difficulty</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Impossible</td>
<td>Insolvable</td>
<td>0</td>
</tr>
<tr>
<td>Low</td>
<td>Strong</td>
<td>1</td>
</tr>
<tr>
<td>Medium</td>
<td>Solvable</td>
<td>2</td>
</tr>
<tr>
<td>High</td>
<td>Easy</td>
<td>3</td>
</tr>
</tbody>
</table>

where, \( N_{\text{ext}} \) is the set of the (external) nodes with no ancestors and represents the graph entry points (attacks). \( N_{\text{ter}} \) is the set of the (terminal) nodes that are the graph end point(s) having no descendants, representing the attackers’ goal(s). Finally, \( N_{\text{int}} \) is the set of the (internal) nodes that have both ancestors and descendants and they represent the vulnerability attributes of potentially insecure network states. The \( i^{th} \) node \( S_i \) is represented by a Bernoulli random variable, i.e. \( S_i \) in \( S \) can be in either true (i.e. \( S_i = 1 \)) or false (i.e. \( S_i = 0 \)) state, meaning, the attacker’s success or failure in reaching such a state, respectively. Accordingly, the probability of node \( S_i \) can be formulated as:

\[
Pr(S_i = 1) = 1 - Pr(S_i = 0) = p, \; p \in \{0, 1\}
\]  

(3.2)

Second, \( E \) is the set of directed edges that connect a child node \( S_i \) to its parent node(s) \( pa[S_i] \). A BAG edge represents the probability of successful exploitation of the \( j^{th} \) network vulnerability \( e_j \), granting attacker(s) access to a network node \( S_i \) from its parent nodes set \( pa[S_i] \). The probability of vulnerability exploitation \( Pr(e_j) \) can be assessed through guesstimating:

i) the attack likelihood \( (Li_j) \), which is a measure of the attacker’s intrinsic characteristics to launch the attack, such as the attacker’s computation power and transmission range requirements. Consequently, \( Li_j \) can vary from impossible to low, medium, or high as shown in Table 3.1.  

ii) The attack impact \( (Im_j) \) which is a measure of the attacker’s gain resulting from potential negative impact on the network. So, \( Im_j \) can vary from no effect to mere annoyance, partial loss of service yet still operational, to complete loss of service, as shown in Table 3.2. Then, each attack is scored on a scale of 0 to 9 according to the product of its
Table 3.2: Impact Component Evaluation Grid [5,31]

<table>
<thead>
<tr>
<th>$Im_j$</th>
<th>Effect</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>No effect</td>
<td>0</td>
</tr>
<tr>
<td>Low</td>
<td>Annoyance</td>
<td>1</td>
</tr>
<tr>
<td>Medium</td>
<td>Partial loss of service</td>
<td>2</td>
</tr>
<tr>
<td>High</td>
<td>Complete loss of service</td>
<td>3</td>
</tr>
</tbody>
</table>

likelihood and impact components [5]. Consequently, $Pr(e_j)$ can be attained by normalizing the product of likelihood and risk components as follows:

$$Pr(e_j) = (L_i_j \times Im_j)/10 \quad (3.3)$$

The method described above is adapted from [89] and was used by the IEEE 802.22 security ad-hoc group in evaluating potential security vulnerabilities of the IEEE 802.22 networks [5]. This is also used by the authors in [31] to evaluate the security threats of different CR design architectures.

**Third**, $\mathcal{R}$ is the set of relations among parent nodes, which can be either logical OR or logical AND relation.

**Fourth**, $\mathcal{P}$ is the set of discrete conditional probability distribution functions for every node $S_i \in N_{int} \cup N_{ter}$. For a node $S_i$, the local conditional probability distribution ($LCPD_i$) represents the probability of an attack to successfully reach node $S_i$ from its parent nodes $pa[S_i]$.

$$LCPD_i = Pr(S_i|pa[S_i]) \quad \forall S_i \in N_{int} \cup N_{ter} \quad (3.4)$$

In accordance to the relation among parent nodes, $Pr(S_i|pa[S_i])$ can be mathematically expressed as:

$$Pr(S_i|pa[S_i]) = \begin{cases} 
0, & \forall S_j \in pa[S_i]|S_j = 0 \\
1 - \prod_{j:S_j}(1 - Pr(e_j)), & otherwise
\end{cases} \quad (3.5)$$
If the relation among parent nodes is logical $OR$ and as:

$$Pr(S_i|pa[S_i]) = \begin{cases} 
0, & \exists S_j \in pa[S_i] \land S_j = 0 \\
\prod_{j:S_j} Pr(e_j), & \text{otherwise}
\end{cases} \quad (3.6)$$

If relation among parent nodes is logical $AND$.

### 3.2.2 Unconditional Probability of Attacking a Node

In BAG model, nodes in set $S$ represent a set of random variables, while edges $E_i$ represent the dependencies among these random variables. Consequently, the chain rule can be used to formulate the joint probability distribution of all nodes as:

$$Pr(S) = \prod_{i=1}^{n} Pr(S_i|pa[S_i]) \quad (3.7)$$

The unconditional probability for node $S_i$ can be obtained by combining all the marginal probabilities at that node.

$$Pr(S_i) = \sum_{S\setminus S_i} \prod_{i=1}^{|S|} Pr(S_i|pa[S_i]) \quad (3.8)$$

where $|S|$ is the cardinality of set $S$ and $S\setminus S_i$ is the sum over all nodes in set $S$ except node $S_i$.

The unconditional probability of external nodes equals to the prior probability of these nodes. There exist two approaches for calculating the prior probability pertaining to the external nodes. The first approach is the probabilistic approach, where a prior probability represents the subjective belief of security experts regarding the associated attack’s existence [42]. This method is sensitive to errors in the very subjective nature of the human estimates or judgment. Besides, most experts tend to give unquantifiable estimations against the attack probabilities, i.e. estimating the attack as rare, likely or most likely to occur [90]. The second
approach for calculating the prior probability is the deterministic approach, where external nodes do not represent random variables, rather they exist deterministically, indicating the attacker’s full privilege over his machine, which is mathematically expressed as:

$$Pr(S_i) = 1 \quad \forall S_i \in N_{ext} \quad (3.9)$$

Intrinsically, the deterministic approach is too conservative and is employed only to derive the worst-case estimation of the surrounding threat environment. In this work, the deterministic approach is extended and used, such that external nodes are represented by binary variables. This is equivalent to the prior probabilities of external nodes in either Unity or Zero to represent the attack status as exist or not, respectively.

$$Pr(S_i) \in \{0, 1\} \quad \forall S_i \in N_{ext} \quad (3.10)$$

In summary, the BAG model inputs are i) the prior probability of external nodes and ii) the probability of vulnerabilities exploitation. While the unconditional probability of internal and terminal BAG nodes are the model output. The BAG output precisely represents the probability of an attacker to reach a BAG node $S_i$ in $N_{int} \cup N_{ter}$.

As an illustration, the computation of the unconditional probabilities of the internal and terminal BAG nodes are depicted using a simplified BAG model representation of two DoS attacks in CR networks shown in Figure 3.1. Node $S_1$: ”DoS for incumbent and policy databases attack” and node $S_2$: ”GPS jamming attack” are external nodes. Node $S_3$: ”authentication failure of incumbent database” and node $S_4$: ”authentication failure of geolocation information” are internal nodes. Node $S_5$: ”attacker denies communication on target frequency channel” is a terminal node. So, to compute the unconditional probability of node $S_5$, we do the following:

i) calculate $Pr(e_j)$ according to the evaluation procedure above as summarized in Table 3.3. For example, $E_1$ represents the probability of $S_1$, as a parent node, to successfully cause $S_3$, as a child node, to exist. In [5] both impact and likelihood of $E_1$ were evaluated as 3. So,
Figure 3.1: Simplified BAG Example

using (3.3), $Pr(e_j)$ is 0.9. ii) Create the LCPD tables of nodes $S_3$, $S_4$ and $S_5$ using (3.5) and (3.6) in accordance to the type of relation among each node’s parents. For example, Node $S_5$ has two parents with logical OR relation between them, so there exist $2^2$ marginal cases to calculate, as depicted by the Tables beside each node in Figure 3.1. iii) Finally, the probability of an attacker to successfully reach node $S_5$ is obtained using (3.8) as follows.

$$Pr(S_5) = \sum_{S_1,S_2,S_3,S_4} Pr(S_1) \times Pr(S_2) \times Pr(S_3|S_1) \times Pr(S_4|S_2) \times Pr(S_5|S_3,S_4)$$

(3.11)

Computing (3.11) yields that $Pr(S_5)$ is approximately equal to 0.595. Similarly, $Pr(S_3)$ and $Pr(S_4)$ can be calculated considering their parents, shown by the red text beside each node in Figure 3.1.
### Table 3.3: Simplified BAG; Evaluation of Edges probabilities [5,31]

<table>
<thead>
<tr>
<th>$E_i$</th>
<th>$S_i$</th>
<th>$pa[S_i]$</th>
<th>$Im_i$</th>
<th>$Li_i$</th>
<th>$P(e_i)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$S_3$</td>
<td>$S_1$</td>
<td>High (3)</td>
<td>High (3)</td>
<td>0.9</td>
</tr>
<tr>
<td>2</td>
<td>$S_4$</td>
<td>$S_2$</td>
<td>Low (1)</td>
<td>High (3)</td>
<td>0.3</td>
</tr>
<tr>
<td>3</td>
<td>$S_5$</td>
<td>$S_3$</td>
<td>Med. (2)</td>
<td>High (3)</td>
<td>0.6</td>
</tr>
<tr>
<td>4</td>
<td>$S_5$</td>
<td>$S_4$</td>
<td>Med. (2)</td>
<td>Med. (2)</td>
<td>0.4</td>
</tr>
</tbody>
</table>

### 3.3 DoS Threat Assessment of the IEEE 802.22 Networks Using BAG Model

In this section, we use the BAG model to assess the DoS probability in the IEEE 802.22 networks in the context of simultaneous multiple attack scenarios.

#### 3.3.1 BAG Model Assumptions

In this work, the following assumptions are considered

1. The probability of vulnerability exploitation is time invariant, as the attacker capabilities barely change with time, such that it can be considered constant. Moreover, this assumption simplifies the threat assessment process.

2. Consider only the adversarial–based DoS threats of the IEEE 802.22 networks that operate in the TV bands. So benign threats such as noise, interference and attenuation are omitted.

3. No negative mutual–effects among attacks within an attack scenario are considered. For example, the negative affect of masking attack on primary user emulation attack, if both are launched at the same time and the same frequency, is neglected. The consideration of such effect is not useful for the security administrative authorities as it may decrease the importance of multiple attackers, which is an aberration from adequate security planning.

Alongside the rationale from the assumptions mentioned above, also they are widely used in
the literature of attack graphs [90].

3.3.2 Building of DoS AG model

In order to create a BAG model representation, we first start with capturing the logical relationships between the attacks, vulnerabilities and the attacks’ goal using the attack graph (AG) model. Typically, the AG model is based upon:

1. The architectural model of the IEEE 802.22 standard [3].
2. The IEEE 802.22 security features and mechanisms [33, 83].
3. The Hammer model of DoS threat assessment [31, 91].
4. The identification and the analysis of CR networks’ vulnerabilities that were discussed in [29, 39] and summarized in Section 3.1.
5. Potential security threats of the IEEE 802.22 that were introduced in [6, 7, 10, 14, 37] and were also pinpointed in Section 3.1.

The proposed AG model representation of the IEEE 802.22 network is explored in Figure 3.2. The AG contains 28 entities that are classified into: i) 17 security threats (red entities), ii) 10 insecure IEEE 802.22 network properties or vulnerabilities (blue entities) and iii) one particular attack goal (a yellow entity). Links \( E_j \) on the AG show the relations among AG entities (thick black arrows), which determine the possible direction of attack evolution. All relationships among parent entities are logical OR unless otherwise depicted by red curves with notation of logical AND.

3.3.3 Building of DoS BAG model

Primarily, the DoS BAG model representation of the IEEE 802.22 networks is created using the pre–created AG model. The DoS BAG is shown in Figure 3.3, such that: i) the BAG nodes are the AG entities: nodes \( \{S_1, S_2, ..., S_{17}\} \in N_{\text{ext}}, \) nodes \( \{S_{18}, S_{19}, ..., S_{27}\} \in N_{\text{int}} \)
and node \( \{S_{28}\} \in N_{\text{ter}} \). ii) the BAG edges are the links among AG entities and iii) \( LCPD_i \). Tables for BAG nodes can be calculated by following the procedure mentioned above after evaluating the BAG edges probabilities \( Pr(e_j) \).

Table 3.4 illustrates the impact \( Im_i \), the likelihood \( Li_i \) and the probability of vulnerability exploitation \( Pr(e_j) \) among child node \( S_i \) and parent node \( S_j \in pa(S_i) \) as captured from [3, 5, 19, 31, 33, 39, 91].

Figure 3.2: The DoS AG of the IEEE 802.22
Lastly, using (3.8), the probability of DoS, \( Pr(S_{28}) \) can be mathematically expressed as:

\[
Pr(S_{28}) = \sum_{S \backslash S_{28}} \Pr(S_1) \times \Pr(S_2) \times \Pr(S_3) \times \Pr(S_4) \times \Pr(S_5) \times \Pr(S_6) \times \Pr(S_7) \times \Pr(S_8) \\
\times \Pr(S_9) \times \Pr(S_{10}) \times \Pr(S_{11}) \times \Pr(S_{12}) \times \Pr(S_{13}) \times \Pr(S_{14}) \times \Pr(S_{15}) \times \Pr(S_{16}) \\
\times \Pr(S_{17}) \times \Pr(S_{18} \mid S_1) \times \Pr(S_{19} \mid S_2) \times \Pr(S_{20} \mid S_3, S_4) \times \Pr(S_{21} \mid S_5) \\
\times \Pr(S_{22} \mid S_6, S_7) \times \Pr(S_{23} \mid S_8, S_9, S_{10}, S_{11}) \times \Pr(S_{24} \mid S_{13}) \times \Pr(S_{25} \mid S_{14}) \\
\times \Pr(S_{26} \mid S_{15}, S_{16}, S_{17}) \times \Pr(S_{27} \mid S_{12}, S_{18}, S_{19}, S_{20}, S_{24}, S_{25}, S_{26}) \\
\times \Pr(S_{28} \mid S_{19}, S_{21}, S_{22}, S_{23}, S_{27})
\] (3.12)

3.3.4 Simultaneous Multiple Attack Scenarios

Broadly, a viable attack scenario can be fully represented by an attack vector \( m \), which contains the prior probabilities of BAG’s external nodes:

\[
m = \{ Pr(S_1), Pr(S_2), ..., Pr(S_{17}) \}
\] (3.13)

Moreover, the prior probabilities of external nodes can be considered as either Unity or Zero to represent whether or not the attack exists, respectively. Define \( p_m \) as the probability
Table 3.4: IEEE 802.22 DoS BAG; Evaluation of Edges probabilities

<table>
<thead>
<tr>
<th>$E_i$</th>
<th>$S_j$</th>
<th>$S_i$</th>
<th>$I_i$</th>
<th>$L_i$</th>
<th>$P(e_i)$</th>
<th>$E_i$</th>
<th>$S_j$</th>
<th>$S_i$</th>
<th>$I_i$</th>
<th>$L_i$</th>
<th>$P(e_i)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$S_1$</td>
<td>$S_{18}$</td>
<td>High (3)</td>
<td>Med. (2)</td>
<td>0.6</td>
<td>2</td>
<td>$S_2$</td>
<td>$S_{19}$</td>
<td>Med. (2)</td>
<td>Med. (2)</td>
<td>0.4</td>
</tr>
<tr>
<td>3</td>
<td>$S_3$</td>
<td>$S_{20}$</td>
<td>High (3)</td>
<td>Low (1)</td>
<td>0.3</td>
<td>4</td>
<td>$S_4$</td>
<td>$S_{20}$</td>
<td>Med. (2)</td>
<td>Med. (2)</td>
<td>0.4</td>
</tr>
<tr>
<td>5</td>
<td>$S_5$</td>
<td>$S_{21}$</td>
<td>Low (1)</td>
<td>High (3)</td>
<td>0.3</td>
<td>6</td>
<td>$S_6$</td>
<td>$S_{22}$</td>
<td>Low (1)</td>
<td>Low (1)</td>
<td>0.1</td>
</tr>
<tr>
<td>7</td>
<td>$S_7$</td>
<td>$S_{22}$</td>
<td>Low (1)</td>
<td>High (3)</td>
<td>0.3</td>
<td>8</td>
<td>$S_8$</td>
<td>$S_{23}$</td>
<td>High (3)</td>
<td>Med. (2)</td>
<td>0.6</td>
</tr>
<tr>
<td>9</td>
<td>$S_9$</td>
<td>$S_{23}$</td>
<td>High (3)</td>
<td>Med. (2)</td>
<td>0.6</td>
<td>10</td>
<td>$S_{10}$</td>
<td>$S_{23}$</td>
<td>High (3)</td>
<td>Low (1)</td>
<td>0.3</td>
</tr>
<tr>
<td>11</td>
<td>$S_{11}$</td>
<td>$S_{23}$</td>
<td>Low (1)</td>
<td>High (3)</td>
<td>0.3</td>
<td>12</td>
<td>$S_{12}$</td>
<td>$S_{27}$</td>
<td>High (3)</td>
<td>Low (1)</td>
<td>0.3</td>
</tr>
<tr>
<td>13</td>
<td>$S_{13}$</td>
<td>$S_{24}$</td>
<td>High (3)</td>
<td>High (3)</td>
<td>0.9</td>
<td>14</td>
<td>$S_{14}$</td>
<td>$S_{25}$</td>
<td>Low (1)</td>
<td>High (3)</td>
<td>0.3</td>
</tr>
<tr>
<td>15</td>
<td>$S_{15}$</td>
<td>$S_{26}$</td>
<td>Low (1)</td>
<td>High (3)</td>
<td>0.3</td>
<td>16</td>
<td>$S_{16}$</td>
<td>$S_{26}$</td>
<td>Med. (2)</td>
<td>Low (1)</td>
<td>0.2</td>
</tr>
<tr>
<td>17</td>
<td>$S_{17}$</td>
<td>$S_{26}$</td>
<td>Med. (2)</td>
<td>Low (1)</td>
<td>0.2</td>
<td>18</td>
<td>$S_{18}$</td>
<td>$S_{27}$</td>
<td>High (3)</td>
<td>High (3)</td>
<td>0.9</td>
</tr>
<tr>
<td>19</td>
<td>$S_{19}$</td>
<td>$S_{27}$</td>
<td>Med. (2)</td>
<td>Low (1)</td>
<td>0.2</td>
<td>20</td>
<td>$S_{20}$</td>
<td>$S_{27}$</td>
<td>Med. (2)</td>
<td>Low (1)</td>
<td>0.2</td>
</tr>
<tr>
<td>21</td>
<td>$S_{21}$</td>
<td>$S_{28}$</td>
<td>Low (1)</td>
<td>High (3)</td>
<td>0.3</td>
<td>22</td>
<td>$S_{22}$</td>
<td>$S_{28}$</td>
<td>Med. (2)</td>
<td>Med. (2)</td>
<td>0.4</td>
</tr>
<tr>
<td>23</td>
<td>$S_{23}$</td>
<td>$S_{28}$</td>
<td>High (3)</td>
<td>Med. (2)</td>
<td>0.6</td>
<td>24</td>
<td>$S_{24}$</td>
<td>$S_{27}$</td>
<td>High (3)</td>
<td>Med. (2)</td>
<td>0.6</td>
</tr>
<tr>
<td>25</td>
<td>$S_{25}$</td>
<td>$S_{27}$</td>
<td>Med. (2)</td>
<td>Med. (2)</td>
<td>0.4</td>
<td>26</td>
<td>$S_{26}$</td>
<td>$S_{27}$</td>
<td>High (3)</td>
<td>High (3)</td>
<td>0.9</td>
</tr>
<tr>
<td>27</td>
<td>$S_{27}$</td>
<td>$S_{28}$</td>
<td>High (3)</td>
<td>High (3)</td>
<td>0.9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

of DoS in the IEEE 802.22 networks when an attack vector $m$ is launched. Hence, $p^n$ can be obtained by computing (3.12) using (3.13) and (3.10). In addition, repeating the former steps for $2^{N_{\text{ext}}}$ times, to consider all possible combinations of prior probabilities, yields corresponding probability of DoS considering all attack scenarios.

3.3.5 Simulation Results and Interpretation

To demonstrate the usefulness of the proposed models, a Matlab–based simulation of the problem of simultaneous multiple attacks is conducted to investigate the probability of DoS in IEEE 802.22 networks considering $2^{17}$ different attack scenarios. Figure 3.4 shows a general view of the results where the attack vectors are depicted on the $x$–axis, while their corresponding DoS probabilities are depicted on the $y$–axis. Notably, the results show that the DoS probability is approximately 91.3% at the most extreme attack scenario, i.e. when the attacker successfully launches 17–DoS attacks.

Moreover, contrary to popular belief on the fixed havoc of sole attacks, such as jamming,
Figure 3.4: The DoS probability of the IEEE 802.22 considering all possible attack scenarios on CRN [5], results show that sole attacks may have different impacts on the CRN according to the targeted vulnerability and the victim CRN’s security defense structure.

A previously published approach where the PUE was estimated as the most extreme sole attack induces approximately 40% probability of DoS in centralized cooperative CR networks architecture [31]. The proposed approach shows that the most extreme simultaneous multiple attack scenario can induce approximately 91.3% probability of DoS in the centralized cooperative IEEE 802.22 networks, which is 51.3% higher than the result for a single attack. It is concluded that considering only single attack grossly underestimates the probability of DoS in the IEEE 802.22 networks.

Broadly, attackers tend to utilize the lowest possible number of effective attacks in order to increase their gain through decreasing their cost [17]. A comparison between attack scenarios can be made according to the number of attacks within each scenario, as depicted in Figure 3.5. It is observed that attack vectors $m_1$, $m_2$, till $m_{17}$, in Figure 3.5, hold the highest DoS probability for each group of attack scenarios. Moreover, Table 3.5 shows the
Figure 3.5: The classification of attack scenarios according to the number of attacks within each scenario

interpretation of $m_1$, $m_2$, till $m_{17}$ referring to the BAG’s external nodes.

Another important observation is that some IEEE 802.22 vulnerabilities (BAG’s internal nodes) are highly exploitable by attackers than some other vulnerabilities. This is captured by Figure 3.6, which shows the normalized aggregate marginal probabilities of BAG’s internal nodes considering all attack scenarios.

Obviously, the impairment of management messages during the transmission of CBS’s final spectrum decision was exploited in approximately 85% of attack scenarios. Also, spoofing the spectrum manager by manipulating i) the on board sensing circuitry and ii) the reception of the spectrum sensing reports was victimized in approximately 40% of attack scenarios. This raises the importance of protecting the cooperative spectrum sensing process being a prime target for the attackers.

Leaning on the above discussion, the BAG model is determinately a single and sufficient model that assesses the effect of different attack scenarios against the CR network.
Table 3.5: IEEE 802.22 DoS BAG; Probable Attack Vectors

<table>
<thead>
<tr>
<th>$m$</th>
<th>$p^m$</th>
<th>$S_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$m_1$</td>
<td>0.486</td>
<td>$S_{13}$</td>
</tr>
<tr>
<td>$m_2$</td>
<td>0.671</td>
<td>$S_{13}, S_8$</td>
</tr>
<tr>
<td>$m_3$</td>
<td>0.750</td>
<td>$S_{13}, S_{12}, S_8$</td>
</tr>
<tr>
<td>$m_4$</td>
<td>0.807</td>
<td>$S_{13}, S_{12}, S_9, S_8$</td>
</tr>
<tr>
<td>$m_5$</td>
<td>0.829</td>
<td>$S_{13}, S_{12}, S_9, S_8, S_7$</td>
</tr>
<tr>
<td>$m_6$</td>
<td>0.842</td>
<td>$S_{13}, S_9, S_8, S_7, S_5$</td>
</tr>
<tr>
<td>$m_7$</td>
<td>0.859</td>
<td>$S_{14}, S_{13}, S_9, S_8, S_7, S_5$</td>
</tr>
<tr>
<td>$m_8$</td>
<td>0.867</td>
<td>$S_{14}, S_{13}, S_{12}, S_9, S_8, S_7, S_5$</td>
</tr>
<tr>
<td>$m_9$</td>
<td>0.875</td>
<td>$S_{14}, S_{13}, S_{12}, S_{10}, S_9, S_8, S_7, S_5, S_2$</td>
</tr>
<tr>
<td>$m_{10}$</td>
<td>0.882</td>
<td>$N_{ext} - {S_1, S_3, S_6, S_{11}, S_{15}, S_{16}, S_{17}}$</td>
</tr>
<tr>
<td>$m_{11}$</td>
<td>0.888</td>
<td>$N_{ext} - {S_3, S_4, S_6, S_{11}, S_{16}, S_{17}}$</td>
</tr>
<tr>
<td>$m_{12}$</td>
<td>0.894</td>
<td>$N_{ext} - {S_3, S_6, S_{11}, S_{16}, S_{17}}$</td>
</tr>
<tr>
<td>$m_{13}$</td>
<td>0.899</td>
<td>$N_{ext} - {S_3, S_6, S_{11}, S_{17}}$</td>
</tr>
<tr>
<td>$m_{14}$</td>
<td>0.903</td>
<td>$N_{ext} - {S_3, S_6, S_{11}}$</td>
</tr>
<tr>
<td>$m_{15}$</td>
<td>0.908</td>
<td>$N_{ext} - {S_3, S_6}$</td>
</tr>
<tr>
<td>$m_{16}$</td>
<td>0.911</td>
<td>$N_{ext} - {S_3}$</td>
</tr>
<tr>
<td>$m_{17}$</td>
<td>0.913</td>
<td>$N_{ext}$</td>
</tr>
</tbody>
</table>

Figure 3.6: The normalized aggregate marginal probabilities of IEEE 802.22 vulnerabilities
3.4 Chapter Summary

In this chapter, the problem of simultaneous multiple DoS attacks in the IEEE 802.22 networks was addressed from the CR network’s viewpoint. The BAG model was utilized to facilitate the assessment process of the problem, providing a feasible metric of CR vulnerabilities. Primarily, potential DoS attacks and known vulnerabilities of the IEEE 802.22 networks were pinpointed. Then, the BAG model was discussed using a supplementary simplified example to illustrate the concept. Finally, the BAG model representation of DoS attacks in the IEEE 802.22 networks was created, making use of the created attack graph that captures the cause–consequences relationship among DoS attacks, vulnerabilities and attacks’ goals.

The Matlab—based simulation results show that multiple attacks can increase the probability of DoS in the IEEE 802.22 networks by up to 51.3%, in comparison to most severe single attacks. Moreover, the impairment of the cooperative spectrum sensing process is the most exploitable CRN’s property considering all attack scenarios. Lastly, the most probable attack scenarios, concerning attacker(s) capabilities, were identified and discussed.
4.1 Introduction

In this chapter, a deception-based defense mechanism is proposed to counteract the multiple coordinated acute jamming attacks in CRNs, named as the deceiving attack henceforth. The aforementioned attack mainly targets the cooperative spectrum sensing process in CRNs. The defense mechanism targets decreasing the deceiving attack’s likelihood in targeting real CRN’s vulnerabilities. The problem is formulated as a Stackelberg security game between the deceiving attacker and the defending CRN. The Stackelberg model is chosen to consider the attacker’s observability of the defender’s actions. The Stackelberg equilibria (SE) are studied under the two cases when the players know and are uncertain about the primary user (PU) activity. Both theoretical analysis and numerical results show that the proposed mechanism can decrease the probability of success of the deceiving attack to nearly 0% when the defender has a high motivation in defending the channel.

\[10^{th}\] The content of this chapter has generated two papers:

Submitted as a manuscript to the IEEE Transactions on Cognitive Communications and Networking [92], I. Ahmed and A. O. Fapojuwo, "Stackelberg equilibria of an anti-jamming game in cooperative cognitive radio networks," received reviewers’ comments in June 2017 and currently being revised for resubmission; June 2017;

4.2 System Model

Mainly due to the adaptability and the dynamic spectrum accessibility, cognitive radio networks (CRNs) are plagued with new security threats besides the traditional threats that are shared with other wireless networks. In its typical form, radio jammers deliberately transmit radio signals to block, mask, or emulate the legitimate active wireless connections.

The deceiving attacker is a radio jammer, equipped with CR platform, which senses the frequency spectrum and launches different types of jamming signals at CRN’s vulnerable times to maximize the impact on the victim CRN.

A deception-based security game $\mathcal{G}$ is between a defender $D$ and an attacker $A$ both of who are the players. In the sequel, the elements of the game are introduced.

4.2.1 The Attacker Model

The first player $A$ is the deceiving attacker whose goal is the denial of CRN’s communication over targeted channel(s). $A$ uses a CR platform and can well distinguish between the signals of PUs and secondary users (SUs). Thus selects her best attack action as follows:

i) during the quiet period, $A$ can launch primary user emulation (PUE) attack, denoted by $l_1$ (i.e., start attack action 1), upon the absence of PU’s signals. If $l_1$ succeeds, unnecessary hand-off for the CRN takes place. Moreover, any future sensing in the evacuated channel is constrained by the non-occupancy-period (NOP)$^{11}$.

ii) During the quiet period, $A$ can launch the masking attack, represented by $l_2$, upon the presence of PU’s signals. $l_2$ may induce a penalty on the victim CRN due to violating the PU rights in conducting an interference-free communication.

iii) During the receiving times of the spectrum reports [the spectrum decision], $A$ can trans-

$^{11}$NOP is the time period where spectrum sensing is not allowed in the evacuated channel when marked as in use by PU. NOP is 10 minutes in the very high frequency/ultra high frequency (VHF/UHF) spectrum for Wireless Regional Area Networks (WRAN) coexistence with DTV [2] and 30 minutes in the 5 GHz spectrum for Wi-Fi coexistence with Radar systems [93].
mit continuous white Gaussian noise to impair the victim’s receiving-circuitry, denoted by \( l_3 \) \( [l_4] \). Since the noise signal transmission is continuous, both the attacks \( l_3 \) and \( l_4 \) are referred to as the \textit{blinding attacks} henceforth.

Denote the maximum number of individual jamming attacks that A can launch by \( L \). Thus, A can launch up to \( L = 4 \) attacks from the preceding. Moreover, let \( Cl_z \) denote the implementation cost of attack \( l_z \), \( 1 \leq z \leq L \) where \( Cl_z = r_z Cl \), \( Cl \) is the attacker’s cost unit and \( r_z \in \mathbb{R}_{>0} \) is the relative cost factor of attack \( l_z \) (i.e. relative to the attacker’s cost unit, \( Cl \)).

Define the \textit{attack strategy} \( m \) as a sequence of \( l_z \), i.e., \( m = \langle l_z \rangle \), \( l_z \in \{0,1\} \), such that \( l_z = 1 \) means attack \( l_z \) is used and 0, otherwise. Let \( \mathcal{M} \) be the set of all attack strategies such that \( \langle m_j \in \mathcal{M} \rangle \) is a possible attack strategy, where \( 1 \leq j \leq |\mathcal{M}| \). In the attacker model with \( L = 4 \) as presented earlier, there are \( |\mathcal{M}| = 2^4 = 16 \) combinations of attacks, but only \textit{six} combinations are viable. The reasons behind the infeasibility of \textit{ten} attack strategies are:

1. \( l_1 \) and \( l_2 \) cannot simultaneously target the same quiet period.

2. The decision on the spectrum access is made through combining the measured sensing information from the onboard sensing circuitry and received sensing reports from the co-operating SUs. Consequently, A launches \( (l_1 \text{ combined with } l_3) \) or \( (l_2 \text{ combined with } l_3) \) upon the absence or the presence of the PU, respectively [31].

To conclude, the feasible attack strategies are: \( m_1 = \{l_1, l_3, l_4\}, m_2 = \{l_1, l_3\}, m_3 = \{l_2, l_3, l_4\}, m_4 = \{l_2, l_3\}, m_5 = \{l_4\} \) and \( m_6 = \{\} = \text{null vector} \) where A prefers not to launch any attacks.

Let \( \Sigma_A = \{\sigma_{m_1}, \sigma_{m_2}, ..., \sigma_{m_{|\mathcal{M}|}}\} \) denote the attacker’s mixed strategy profile which specifies the probability assigned to each attack strategy, where \( \sigma_{m_j} \in [0,1] \) is the probability assigned to attack strategy \( m_j \) and \( \sum_{j=1}^{|\mathcal{M}|} \sigma_{m_j} = 1 \) by the total probability theorem. The attacker’s pure strategy profile is a special case of \( \Sigma_A \) in which all \( \sigma_{m_j} \in \Sigma_A = \{0,1\} \).
4.2.2 The Defender Model

Without loss of generality and to introduce a strong case study, the IEEE 802.22 CR network is used as the defender model\(^{12}\). Integrating the proposed defense mechanism with the standard security features of the IEEE 802.22 enables the transmission of deceptive signals over the channel by CBS/SUs at scheduled times with a goal to trigger A to attack. The attacker A cannot distinguish between the real signals and those of the deceptive signals (honeypot). If A attacks a honeypot, first, the attack is ineffective because it did not target a real vulnerability, meaning that A has wasted some of her resources (such as stored energy) without causing any impact on the network. Second, the victim CBS/SUs can detect the direction-of-arrival (DoA) of jamming signals\(^{13}\), thus place jammer’s direction in the nulls of the antenna using any of the adaptive nulling antenna (ANA)-based techniques, such as those introduced in [57–59].

The defender D utilizes up to N honeypots, where the \(n^{th}\) honeypot is denoted by \(k_n\), \(n = 1, 2, ..., N\). Each honeypot \(k_n\) is uniquely designed to protect a distinct CRN vulnerability, meaning the number of available honeypots \(N\) is equal to the number of CRN known vulnerabilities that are required to be protected. From chapter 3, key CRN known vulnerabilities are threefold (i.e., \(N = 3\)) listed as follows:

1. The quiet period (QP).

2. The transmission of the spectrum sensing reports by the cooperating SUs

3. The broadcast of the final decision regarding the spectrum access by the CBS.

\(^{12}\) Besides being the first complete standard that utilizes CR technology in the area of wireless communication, the IEEE 802.22 supports solid security features which help the study/development of the proposed security mechanism [5,33].

\(^{13}\) According to the IEEE 802.22 standard, the SUs and the CBS are required to be aware of their geolocation [2]. Also, the SUs gain the knowledge on the honeypots’ schedule (and the quiet period schedule) when they initially join the CRN after the secured authentication procedure takes place [2].
The CRN vulnerability 1, 2, and 3 are protected by honeypots $k_1$, $k_2$ and $k_3$, respectively. In $k_1$, the CBS sends benign-emulated PU signals over the sensed channel to deceive the PUE attacker who is sensing the channel. $k_1$ highly affects the attacker’s behavior as $A$ cannot differentiate between the legitimate PU signals and the benignly emulated PU signals. If $A$ targets $k_1$ with the masking attack, the attacked CBS/SUs can exclude the defined attacker’s direction from the antenna pattern, returning a penalty $\phi$ on $A$ (also referred to as the moving cost in this thesis) as she needs to relocate the identified malicious platform to retain her attack capabilities.

In $k_2$, one or more other transmission(s) of the spectrum reports are scheduled and utilized as honeypots. Thus, if the blinding attack targets $k_2$, the CBS can identify the direction of the jamming attacker, again returning $\phi$ on the attacker.

Honeypot $k_3$ is in the form of one or more transmission(s) of the spectrum decision by the CBS. Similarly, SUs can identify the direction of the blinding attack if it targets $k_3$. Notice that the effectiveness of the proposed security mechanism when small (or no) penalty $\phi$ on the attacker can be imposed is discussed later in Section 4.5.

Define $D$’s deception strategy ($h$) as a sequence of $k_n$, i.e., $h = \langle k_n \rangle$ where $1 \leq n \leq N$, $k_n \in \{0, 1\}$, such that $k_n = 1$ means honeypot $k_n$ is used and $k_n = 0$, otherwise. Let $\mathcal{H}$ be the set of all possible deception strategies such that $< h_i \in \mathcal{H} >$ is a possible deception strategy, where $1 \leq i \leq |\mathcal{H}| = 2^N$. For the deception model described above with $N = 3$, the cardinality of the deception strategies is 8. Principally, player $D$ schedules honeypot $k_1$ combined with $k_2$ and $k_3$. Because $k_1$ attains no spectrum sensing, consequently, no spectrum reports or spectrum decision are transmitted by SUs or CBS. Instead, $k_2$ and $k_3$ are to be used to mimic the real sensing cycle completely, thus attracts the attacker. The preceding reduces the number of the deception strategies from 8 to 5, enumerated as follows: $h_1 = \{k_1, k_2, k_3\}$, $h_2 = \{k_2\}$, $h_3 = \{k_3\}$, $h_4 = \{k_2, k_3\}$ and $h_5 = \{\} = \text{null vector}$.

Similar to $A$, $\Sigma_D = \{\sigma_{h_1}, \sigma_{h_2}, \ldots, \sigma_{h_{|\mathcal{H}|}}\}$ is the defender’s mixed strategy profile where
\( \sigma_{h_i} \in [0, 1] \) and \( \sum_{i=1}^{\vert H \vert} \sigma_{h_i} = 1 \). The defender’s pure strategy profile is a special case of \( \Sigma_D \) in which all \( \sigma_{h_i} \in \Sigma_D = \{0, 1\} \).

Moreover, let \( Ck_n \) denote the cost incurred by \( D \) when deploying \( k_n \). For example, \( Ck_n \) may represent a reduction in the useful communication time of \( D \) when honeypot \( k_n \) is deployed. In particular, \( Ck_n = q_n Ck \) where \( Ck \) is the defender’s cost unit and \( q_n \in \mathbb{R}_{>0} \) is the relative cost factor of honeypot \( k_n \) (i.e., relative to the defender’s cost unit, \( Ck \)).

Figure 4.1(a) shows a conventional frame structure in cooperative CRNs where \( t_1, t_2, \) and \( t_3 \) represent the times required for i) sensing the spectrum, ii) sending the sensing reports by
the SUs and iii) sending the spectrum decision by the CBS, respectively. Period $t_4$ represents the useful communication time.

Figure 4.1(b) shows the frame structure when the honeypots are utilized. To completely deceive the observing attacker, let $Ck_1 = t_1, Ck_2 = t_2, Ck_3 = t_3$, meaning, the period of deploying a honeypot $k_n$ equals the period of its associated vulnerability $n$. Notice that the CBS is assumed to share the honeypots’ schedule with the SUs within its cell and the nodes in the neighboring CRNs. The sharing of the honeypots’ settings among co-operating entities can be studied in the extension of this work.

4.2.3 The Payoff Functions and the Normal Form

It is quite common in the literature of security games to formulate players’ utility functions as a minimization of the total-loss in a zero-sum framework [45]. The reason is to represent the strictly competing nature of the players, yet assuming each player’s gain/loss is equitable to other player’s gain/loss. Another reason for such a formulation is to represent how the defender minimizes the cost of its deployed deception actions (honeypots) and also how the attacker reduces the loss due to its interaction with the deception actions. Nevertheless, in this work, players’ objectives are still conflicting, but players’ gains/costs are not assumed to be precisely balanced, thus forming a general-sum game instead of a zero-sum game.

Furthermore, the players’ utility functions are formulated as a maximization of payoff functions where, on the one hand, the defender gains more from capturing the attacker by implementing more honeypots. On the other hand, the attacker gains more by avoiding the deployed honeypots. This formulation is more realistic as it provides the flexibility to consider independent defense and attack incentives in different game scenarios.

Primarily, when $A$ launches $m_j$ while $D$ is deploying $h_i$, some malicious actions directly fall into deployed honeypots while others do not. Consequently, players’ gains can be formulated as:

$$G_A(i,j) = (1 - p_{\phi}^{(i,j)}) \ast p_s^{(j)} \ast U_A$$  \hspace{1cm} (4.1a)
\[ G_D(i, j) = p_{\phi}^{(i,j)} \ast U_D \]  

where \( G_A(i, j) \) and \( G_D(i, j) \) are \( A \)'s and \( D \)'s expected gains, respectively and \( p_{\phi}^{(i,j)} \) is the probability of attack actions \( l_z \in m_j \) falling into honeypots \( k_n \in h_i \),  
\[ z = 1, 2, \ldots, L, \quad j = 1, 2, \ldots, 2^L, \quad n = 1, 2, \ldots, N, \quad i = 1, 2, \ldots, 2^N. \]  
And \( p_s^{(j)} \) is the probability of the success of attack strategy \( m_j \) when not falling into honeypots and \( U_A \) is a positive constant which represents \( A \)'s return (i.e., gain) when capturing the channel. \( U_D \) is a positive constant which represents the importance of capturing the attacker, therefore conducting a useful communication over a channel. The constant \( U_D \) could be related to the abundance of free channels in the spectrum, e.g. when the available (free) channels are limited, \( D \) retains a higher gain [loss] when conducting [not conducting] communication over the targeted channel. Similarly, players' losses are:

\[ \Phi_A(i, j) = p_{\phi}^{(i,j)} \ast \phi \]  

\[ \Phi_D(i, j) = (1 - p_{\phi}^{(i,j)}) \ast p_s^{(j)} \ast U_D \]  

where \( \Phi_A(i, j) \) is \( A \)'s expected loss due to falling into honeypots and \( \phi \) is the cost of relocating the identified attacker's platform. In the case of a malicious SU, \( \phi \) may represent the penalty applied by the CRN on the misbehaving SU, such as bandwidth limitation or halting communication for a period of time. \( \Phi_D(i, j) \) is \( D \)'s expected loss due to not protecting attacked vulnerabilities. Then, a player’s payoff function is the difference between the player’s gain and loss as follows:

\[ \Omega_A(i, j) = G_A(i, j) - \Phi_A(i, j) - Cm_j \]  

\[ \Omega_D(i, j) = G_D(i, j) - \Phi_D(i, j) - Ch_j \]  

where \( \Omega_A(i, j) \) and \( \Omega_D(i, j) \) are \( A \)'s and \( D \)'s payoff function, respectively. And \( Cm_j = \sum_{l_z \in m_j} Cl_z \) is the cost of implementing attack strategy \( m_j \). Finally, \( Ch_i = \sum_{k_n \in h_i} Ck_n \) is the cost of implementing deception strategy \( h_i \).

In the sequel, game \( \mathcal{G} \) is described in a bi-matrix (normal form) for the two cases of the PU activities: PU is not using the channel and PU is using the channel. Using the
Table 4.1: The normal form of game $G$ when the PU is not using the channel

<table>
<thead>
<tr>
<th>$b_1$</th>
<th>$b_2$</th>
<th>$b_3$</th>
<th>$b_4$</th>
<th>$b_5$</th>
<th>$b_6$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p^0_s U_D - (1 - p^0_s)p^0_s U_D - Ch_1$</td>
<td>$p^0_s U_D - (1 - p^0_s)p^1_s U_D - Ch_2$</td>
<td>$p^0_s U_D - (1 - p^0_s)p^0_s U_D - Ch_3$</td>
<td>$p^0_s U_D - (1 - p^0_s)p^1_s U_D - Ch_4$</td>
<td>$p^0_s U_D - (1 - p^0_s)p^1_s U_D - Ch_5$</td>
<td></td>
</tr>
<tr>
<td>$-p^0_s \phi - \phi - Cm_1$</td>
<td>$(1 - p^0_s)p^0_s U_A - p^1_s \phi$</td>
<td>$-p^0_s \phi - \phi - Cm_3$</td>
<td>$-p^0_s \phi - \phi - Cm_3$</td>
<td>$-p^0_s \phi - \phi - Cm_3$</td>
<td></td>
</tr>
</tbody>
</table>

Table 4.2: The normal form of game $G$ when the PU is using the channel

<table>
<thead>
<tr>
<th>$b_1$</th>
<th>$b_2$</th>
<th>$b_3$</th>
<th>$b_4$</th>
<th>$b_5$</th>
<th>$b_6$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p^0_s U_D - (1 - p^0_s)p^1_s U_D - Ch_1$</td>
<td>$p^0_s U_D - (1 - p^0_s)p^1_s U_D - Ch_2$</td>
<td>$p^0_s U_D - (1 - p^0_s)p^0_s U_D - Ch_3$</td>
<td>$p^0_s U_D - (1 - p^0_s)p^0_s U_D - Ch_4$</td>
<td>$p^0_s U_D - (1 - p^0_s)p^1_s U_D - Ch_5$</td>
<td></td>
</tr>
<tr>
<td>$(1 - p^0_s)p^1_s U_A - p^0_s \phi$</td>
<td>$-p^1_s \phi - \phi - \phi - Cm_2$</td>
<td>$-p^1_s \phi - \phi - \phi - Cm_2$</td>
<td>$-p^1_s \phi - \phi - \phi - Cm_2$</td>
<td>$-p^1_s \phi - \phi - \phi - Cm_2$</td>
<td></td>
</tr>
</tbody>
</table>

six feasible attack vectors and the five feasible deception vectors identified for the CRN, Table 4.1 and Table 4.2 show 30 different combinations (i.e. 6 attack vectors x 5 deception vectors) of players' strategies and corresponding payoffs when the PU is not using the channel and when the PU is using the channel, respectively.

Within each cell of Tables 4.1 and 4.2, the top expression represents $D$'s payoff $\Omega_D(i, j)$ and the bottom formula is $A$'s payoff $\Omega_A(i, j)$, calculated using (4.3a) and (4.3b), respectively. Note that the PUE and the jamming attacks lead to the same impact on the CR network when targeting the victim while sensing the channel and the PU is using the chan-
nel. Moreover, the masking attack holds no effect if launched while the PU is not using the channel.

Tables 4.1 and 4.2 show players’ interaction over one communication channel. However, it is easy to extend the game to the case when players interact over multiple channels instead of a single channel, done by changing the attacker’s (defender’s) strategy space to include different actions over the channels available in the spectrum.

\[ \binom{x}{y} \times Q \]

\[ Q \] is the number of defense (attack) strategies.

Footnote 14: For example, if the CRN (the attacker) can defend (attack) \( x \) frequency channels out of \( y \) available frequency channels, the number of bi-matrix argument from each player’s side will be \( \binom{x}{y} \times Q \) where \( Q \) is the number of defense (attack) strategies.
4.3 The IEEE 802.22 Stackelberg Deception-based Game Problem

In the Stackelberg model, player $D$ commits to her strategy before player $A$ does. Therefore, $A$’s response can be considered as a function $g(\Sigma_D)$ that maps $\Sigma_D \to \Sigma_A$. Moreover, each $\Sigma_D$ induces a sub-game for $A$ which is solved to find $\Sigma_A$. Thus, $A$’s problem can be mathematically expressed as follows [94]:

\[
\mathcal{P}_1 : \quad \max_{\sigma_{m_j}} \sum_{i=1}^{\|H\|} \sum_{j=1}^{\|M\|} \sigma_{h_i} \sigma_{m_j} \Omega_A^{(i,j)} \tag{4.4a}
\]

subject to:
\[
\sum_{j=1}^{\|M\|} \sigma_{m_j} = 1, \quad \sigma_{m_j} \in [0, 1] \tag{4.4b}
\]

It is clear from (4.4) that the maximum of the attacker’s payoff in (4.4a) is attained by setting $\sigma_{m_j} = 1$ for the $j$-th coefficient that holds the highest value for $\Omega_A^{(i,j)}$. So, there always exists a pure attack strategy for $A$ to solve problem $\mathcal{P}_1$. Thus, $D$’s problem ($\mathcal{P}_2$) is:

\[
\mathcal{P}_2 : \quad \max_{\Sigma_D} \Omega_D^{\Sigma_D,g(\Sigma_D)} \tag{4.5a}
\]

subject to:
\[
\forall \Sigma_D, g' : \quad \Omega_D^{\Sigma_D,g(\Sigma_D)} \geq \Omega_A^{\Sigma_D,g'(\Sigma_D)} \tag{4.5b}
\]
\[
\forall \Sigma_D : \quad \Omega_D^{\Sigma_D,g(\Sigma_D)} \geq \Omega_D^{\Sigma_D,\eta(\Sigma_D)} \tag{4.5c}
\]
\[
\sigma_{m_j} \in \{0, 1\}, \quad \sum_{j=1}^{\|M\|} \sigma_{m_j} = 1 \tag{4.5d}
\]
\[
\sigma_{h_i} \in [0, 1], \quad \sum_{i=1}^{\|H\|} \sigma_{h_i} = 1 \tag{4.5e}
\]
where $\Omega^{\Sigma D; g(\Sigma D)} = \sum_{i=1}^{\vert H \vert} \sum_{j=1}^{\vert M \vert} \sigma_{h_i} \sigma_{m_j} \Omega^{(i,j)}$. The objective function (4.5a) maximizes $D$’s expected payoff given $A$’s best response $g(\Sigma D)$, while the first constraint (4.5b) states that $A$ observes $\Sigma D$ and plays $g(\Sigma D)$ optimally. The second constraint (4.5c) specifies that $A$ breaks ties for $D$, meaning, if $A$ has many best responses to $\Sigma D$, she selects the one that maximizes $D$’s payoff and $\eta$ is the set of $A$’s best responses. The third constraint (4.5d) states that $A$ is only playing pure strategy profiles. Finally the last constraint (4.5e) indicates that $D$ is playing mixed strategy profiles. Note that, there always exist a pure attack strategy and a mixed deception strategy that solve $\mathcal{P}1$ and $\mathcal{P}2$, respectively [95].

### 4.3.1 Guesstimating Game Parameters

In this subsection, *three* game parameters are calculated in the context of the CRN security attributes of *four* attack actions, *six* feasible attack strategies, *three* deception actions (honeypots) and *five* deception strategies, as described in Section 4.2. The game parameters are attacks’ relative cost factors $\{r_1, ..., r_4\}$, the probability of success of attack strategies $p_s^{(j)}$ and the probability of falling into honeypots $p_\phi^{(i,j)}$.

#### 4.3.1.1 Attacks’ relative cost factors

The values of $\{r_1, ..., r_4\}$ can be assessed by guesstimating the technical difficulties, faced by $A$ when implementing PUE, masking and blinding attacks. The complexity of jamming attacks in CRNs was evaluated in [5,18,31] while that in WiMax was reported in [96], where the result of attack complexity evaluation ranges from 0 (i.e., None), 1 (i.e., Easy), 2 (i.e., Solvable) to 3 (i.e., Strong) [5,18,31]. Applying the evaluation results in the aforementioned works, the value of $r_1$ representing the complexity (i.e., relative cost factor) of PUE attack is set to 2 (i.e., Solvable) as it requires the generation of a signal with specific characteristics, e.g. Digital TV signals. The values of $r_2, r_3,$ and $r_4$, representing the complexity of other jamming attacks are set to 1 (i.e., Easy) because each requires the generation of continuous white Gaussian noise.
For instance, the cost of attack strategies \( m_3 = \{l_1, l_3, l_4\} \) is calculated as

\[
C_{m_3} = \sum_{l_z \in m_3} r_z C_l = r_1 C_l + r_3 C_l + r_4 C_l = 4C_l
\]  
(4.6)

Similarly, the cost of all attack strategies are:

\[
C_{m_1} = 3C_l, \quad C_{m_2} = 2C_l, \quad C_{m_3} = 4C_l, \quad C_{m_4} = 3C_l, \quad C_{m_5} = C_l, \quad \text{and} \quad C_{m_6} = 0
\]  
(4.7)

It is important to emphasize that the attacker’s cost unit \( Cl \) is used as a design parameter when discussing the analytic results in this section and the numerical results in Section 4.5.

4.3.1.2 Probability of Success of Attack Strategy \( m_j \)

The probability of success of attack strategy \( m_j \), denoted by \( p_{(j)} \), is calculated using the Bayesian attack graph (BAG) model as introduced earlier in Chapter 3 [18], where the BAG model was used to provide a quantifiable measure of the probability of success of multiple simultaneous DoS attacks in the IEEE-802.22 CRNs.

Figure 4.2 shows the developed BAG model for the deceiving attack in the IEEE 802.22 CRN, where nodes \( S_1, S_2, S_3 \) and \( S_4 \) each represents attack action \( l_1, l_2, l_3 \) and \( l_4 \), respectively. Node \( S_5 \) represents the attacker’s goal of attack actions \( (S_1 \text{ or } S_2), S_3 \text{ and } S_4 \) which is to partially/completely forbid CRN’s communication over a specific/entire frequency channel/band. In the BAG model, a node that launches an attack action is referred to as a parent node while the node that serves as the goal of the attack is the child node.

From Figure 4.2, nodes \( S_1, S_2, S_3 \) and \( S_4 \) are the parent nodes while node \( S_5 \) is the child node. All relations among parent nodes are logical OR unless otherwise depicted on the BAG. The logical AND relation on the BAG means that fusion of the spectrum decision takes place upon receiving both the sensing information from the onboard sensing circuitry and the sensing reports from the co-operating SUs. The logical NOT relation on the BAG means that either node \( S_1 \) or \( S_2 \) exists at any point in time. The edges \( e_1, e_2, e_3 \) and \( e_4 \) on the graph each represents a possible direction of launching an attack action from the parent node(s) \( S \in \{S_1, S_2, S_3, S_4\} \) to the child node \( S_5 \). Each parent node \( S \in \{S_1, S_2, S_3, S_4\} \) can
be in either true (i.e. $S = 1$) or false (i.e. $S = 0$) state, representing whether or not the
attack action(s) $l_z$, $z = 1, 2, 3, 4$ is (are) launched, respectively.

The probability of vulnerability exploitation $Pr(e_z)$ is the weight attached to each edge
$e_z$, which captures both the negative impact and likelihood of launching attack action $l_z$,
denoted by $Im_z$ and $Li_z$, respectively. Both $Im_z$ and $Li_z$ are each scored on a scale from
0 to 3 (i.e., $\{Im_z, Li_z\} \in \{0, 1, 2, 3\}$) and the digits 0, 1, 2 and 3 represent no, low, medium
and high impact/likelihood, respectively. Next, $Pr(e_z)$ is calculated based on the expected
negative impact $Im_z$ and likelihood $Li_z$ of each attack where $Pr(e_z) = (Li_z \times Im_z)/10$ [5].
Table 4.3 provides the assumed values for $Li_z$ and $Im_z$ for the four attack actions, based on
which the values of $Pr(e_z)$ are calculated. Algorithm-1 is then run to calculate $p_s^{(j)}$ as:

$$p_s^{(1)} = 0.37, \quad p_s^{(2)} = 0.1, \quad p_s^{(3)} = 0.43, \quad p_s^{(4)} = 0.18, \quad p_s^{(5)} = 0.3, \quad p_s^{(6)} = 0.0 \quad (4.8)$$
Table 4.3: Evaluation of edge probability in the BAG model [5,31]

<table>
<thead>
<tr>
<th>$e_z$</th>
<th>$Im_z$</th>
<th>$Li_z$</th>
<th>$P(e_z)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>High (3)</td>
<td>Med. (2)</td>
<td>0.6</td>
</tr>
<tr>
<td>2</td>
<td>Low (1)</td>
<td>High (3)</td>
<td>0.3</td>
</tr>
<tr>
<td>3</td>
<td>Low (1)</td>
<td>High (3)</td>
<td>0.3</td>
</tr>
<tr>
<td>4</td>
<td>Low (1)</td>
<td>High (3)</td>
<td>0.3</td>
</tr>
</tbody>
</table>

Algorithm 1: The calculation of the probability of success of attack strategy $m_j$

**Input:** The BAG model representation of the IEEE 802.22 network and the deceiving attack

**Output:** $p_s^{(j)}$; the probability of success of attack strategy $m_j$

1: for all $m_j$, $j \in \{1 : 6\}$ do
2: $S_z = \{0, 1\}^z \ \forall l_z \in m_j$ and $S_z \in pa(S_5)$, where $pa(S_5) = \{S_1, S_2, S_3, S_4\}$
3: for all parents of node $S_5$ do
4: if the relation among parents is logical OR: then
5: \[
Pr(S_5|pa(S_5)) = \begin{cases} 
0, & \forall S_z \in pa(S_5) = 0 \\
1 - \prod_{z=1:4} (1 - Pr(e_z)), & otherwise 
\end{cases} \tag{4.9}
\]
6: else
7: The relation among parents is logical AND:
\[
Pr(S_5|pa(S_5)) = \begin{cases} 
0, & \exists S \in pa(S_5) = 0 \\
\prod_{z=1:4} Pr(e_z), & otherwise 
\end{cases} \tag{4.10}
\]
8: end if
9: end for
10: $p_s^{(j)} = \sum_{S \setminus S_5} \prod_{S} Pr(S_5|pa(S_5)) \tag{4.11}$
11: end for
In Algorithm 1, for each attack strategy \( m_j \), \( j \in \{1 : 6\} \) the prior probabilities of each external BAG node \( S_z \in pa(S_5) \), where \( pa(S_5) = \{S_1, S_2, S_3, S_4\} \) is set to either Unity or Zero according to the status of the associated jamming action \( l_z \in m_j \) being in existence or not, respectively (line 2). Then, the conditional probability distribution function \( Pr(S_5|pa(S_5)) \) is calculated according to the relation among the parents of node \( S_5 \) (line 3). Finally, Algorithm 1 calculates the unconditional probability \( p^{(i)}_s \) of node \( S_5 \) when attack strategy \( m_j \) is launched (line 10). Section 3.2 provides more details on the BAG model.

The computation complexity of running Algorithm-1 is \( \mathcal{O}(2^{|S|}) \) with \( S \) being the set of all nodes on the BAG model and the cardinality \( |S| \) is equal to 5 from Figure 4.2.

4.3.1.3 Probability of falling into honeypots
Recall that the attacker cannot distinguish between the honeypot signals and the legitimate signals. Also, every honeypot is independently designed to attract a specific type of jamming attacks as illustrated in Section 4.2. So, let \( p_{\phi}(k_n|l_z) \) denote the probability of falling into a honeypot \( k_n \) given attack action \( l_z \) is launched, which is calculated by:

\[
p_{\phi}(k_n|l_z) = \frac{Ck_n}{(Ck_n + t_n)}
\] (4.12)

where \( t_n \) is the time period of vulnerability \( n \) and \( Ck_n \) is the time period of honeypot \( k_n \), as shown earlier in Figure 4.1. By definition, \( Ck_n = t_n \) to fully deceive the attacker, thus \( p_{\phi}(k_n|l_z) = 0.5 \). Then, the probability of falling into a honeypot \( k_n \in h_i \) of an attack action \( l_z \in m_j \) can be mathematically expressed as:

\[
p_{\phi}^{(i,j)} = 1 - \prod_{n=1:3} (1 - p_{\phi}(k_n|l_z))
\] (4.13)

Typically, \( p_{\phi}^{(i,j)} \in \{0.5, 0.75, 0.875\} \), representing the probability of one, two, or three attack actions \( l_z \in m_j \) falling into one, two, or three honeypots \( k_n \in h_i \), respectively. Without loss of generality, it is assumed that the value of probability of one attack action falling into one honeypot is identical for all \( i \) and all \( j \). Hence, the superscript \((i, j)\) in \( p_{\phi}^{(i,j)} \)
is replaced by (1) to give $p^{(1)} = 0.5$, for notational simplicity. The notational simplicity is also applied to the probability of two attack actions falling into two honeypots and the probability of three attack actions falling into three honeypots. Hence, in the rest of this thesis, $p^{(i,j)}_\phi$ is then expressed as:

$$p^{(1)}_\phi = 0.5, \quad p^{(2)}_\phi = 0.75, \quad p^{(3)}_\phi = 0.875 \quad (4.14)$$

In the sequel, the game problem in (4.5) is solved using the backward induction method where $D$ solves $A$’s problem first (backward in time) before calculating its response [97]. $G$ is solved first for the special case when $D$ is playing pure strategy profiles and the PU activity pattern is common knowledge in the game, yielding a game with complete information. Then, $G$ is solved for the general case, when $D$ plays mixed strategy profiles and players are uncertain about the game outcome because of the uncertainty about the PU activities, yielding a Bayesian Stackelberg game.

The following definitions are required before proceeding with the solution:

1. $Z^{(i,j)}_A = \Omega^{(i,j)}_A / C_l$ is $A$’s Normalized Payoff function.

2. $Z^{(i,j)}_D = \Omega^{(i,j)}_D / C_k$ is $D$’s Normalized Payoff function.

3. $I_A = U_A / C_l$ is $A$’s incentive factor which represents $A$’s motivation in capturing the channel due to attacking activities.

4. $I_D = U_D / C_k$ is $D$’s incentive factor which represents $D$’s motivation in capturing the channel due to defending activities.

5. $T_A = \phi / C_l$ is $A$’s deterrent factor.

ii) Moreover, let $\{I_A, I_D, T_A\} \in \mathbb{R}_{\geq 1}$ mean that $I_A$, $I_D$ and $T_A$ each assumes values greater than or equal to unity and thus fixes the lower limit of the player’s incentives and deterrent factors to be at least the cost unit of the attack/defense action, to simplify the analysis.
4.3.2 The Special Case: A Game with Complete Information

4.3.2.1 When the PU is not using the channel

A’s best responses are calculated using Table 4.1 and Eqns. (4.7) and (4.8) as follows:

1. When \( D \) plays \( h_1 \):

\[
\max_j Z_A^{(1:j)} = \max_j \left[ \sigma_{m1} Z_A^{(1:1)} + \sigma_{m2} Z_A^{(1:2)} + \sigma_{m3} Z_A^{(1:3)} + \sigma_{m4} Z_A^{(1:4)} + \sigma_{m5} Z_A^{(1:5)} + \sigma_{m6} Z_A^{(1:6)} \right]
\]

where \( j \) is the index of \( A \)’s attack strategy, \( Z_A^{(i:j)} = \Omega_A^{(i:j)}/C_l \), \( I_A = U_A/C_l \) and \( T_A = \phi/C_l \). Then, using Table 4.1 and Eqns. (4.7) and (4.8)

\[
\max_j Z_A^{(1:j)} = \max_j \left[ \sigma_{m1} ((1 - p_\phi^{(3)}) p_s^{(5)} I_A - p_\phi^{(3)} C_m/C_l) + \sigma_{m2} ((1 - p_\phi^{(3)}) p_s^{(5)} I_A - p_\phi^{(3)} C_m/C_l) + \sigma_{m3} ((1 - p_\phi^{(3)}) p_s^{(5)} I_A - p_\phi^{(3)} C_m/C_l) + \sigma_{m4} ((1 - p_\phi^{(3)}) p_s^{(5)} I_A - p_\phi^{(3)} C_m/C_l) + \sigma_{m5} ((1 - p_\phi^{(3)}) p_s^{(5)} I_A - p_\phi^{(3)} C_m/C_l) + \sigma_{m6} (0) \right]
\]

It is clear that \( \max_j Z_A^{(1:j)} \) is at: \( i) \) \( \Sigma_A = \{0,0,0,0,1\} \) for \( F_1 = True \), or \( ii) \) \( \Sigma_A = \{0,0,0,1,0\} \) for \( F_1 = False \) and \( F_2 = True \) , or \( iii) \) \( \Sigma_A = \{0,0,1,0,0\} \) otherwise, where \( F_1 \) is \( True \) if \( I_A < \frac{p_\phi^{(1)} C_m (C_m/C_l)}{1-p_\phi^{(1)}} \) and \( F_2 \) is \( True \) if \( I_A < \frac{p_\phi^{(1)} C_m (C_m/C_l)}{1-p_\phi^{(1)}} \).

2. When \( D \) plays \( h_2 \):

\[
\max_j Z_A^{(2:j)} = \max_j \left[ \sigma_{m1} ((1 - p_\phi^{(1)}) p_s^{(5)} U_A - p_\phi^{(1)} C_m) + \sigma_{m2} ((1 - p_\phi^{(1)}) p_s^{(5)} U_A - p_\phi^{(1)} C_m) + \sigma_{m3} ((1 - p_\phi^{(1)}) p_s^{(5)} U_A - p_\phi^{(1)} C_m) + \sigma_{m4} ((1 - p_\phi^{(1)}) p_s^{(5)} U_A - p_\phi^{(1)} C_m) + \sigma_{m5} (p_s^{(5)} U_A - C_m) + \sigma_{m6} (0) \right]
\]

So, \( \max_j Z_A^{(2:j)} \) is at: \( i) \) \( \Sigma_A = \{0,0,0,0,1\} \) for \( F_3 = True \) , or \( ii) \) \( \Sigma_A = \{0,0,0,1,0\} \) for \( F_4 = False \) and \( F_2 = True \) , or \( iii) \) \( \Sigma_A = \{0,0,1,0,0\} \)
otherwise, where $F_3$ is True if $I_A < \frac{Cm_3/Cm_1}{p_5^{(3)}}$, $F_4$ is True if

$$I_A < \frac{p_{\phi}^{(1)}T_A + (Cm_3 - Cm_5)/Cm_1}{p_5^{(1)}(1-p_5^{(1)}) - p_5^{(3)}}.$$  

3. Similarly, when $D$ plays $h_3$: $\max_j Z^{(3;j)}_A$ is at: 
   i) $\Sigma_A = \{0, 0, 0, 0, 0, 1\}$ for $F_5 = True$ and $F_6 = True$, or
   ii) $\Sigma_A = \{0, 0, 0, 0, 1, 0\}$ for $F_6 = False$ and $F_7 = True$, or
   iii) $\Sigma_A = \{0, 0, 1, 0, 0, 0\}$ for $F_8 = True$, or
   iv) $\Sigma_A = \{0, 0, 0, 1, 0, 0\}$, otherwise where $F_5$ is True if $I_A < \frac{Cm_4/Cm_1}{p_4^{(5)}}$, $F_6$ is True if $I_A < \frac{p_{\phi}^{(1)}T_A + Cm_5/Cm_1}{p_5^{(1)}(1-p_5^{(1)})}$, $F_7$ is True if $I_A < \frac{p_{\phi}^{(1)}T_A + (Cm_3 - Cm_5)/Cm_1}{p_5^{(1)}(1-p_5^{(1)}) - p_5^{(3)}}$ and $F_8$ is True if $I_A < \frac{p_{\phi}^{(1)}T_A + (Cm_3 - Cm_4)/Cm_1}{p_5^{(1)}(1-p_5^{(1)}) - p_5^{(4)}}$.

4. When $D$ plays $h_4$: $\max_j Z^{(4;j)}_A$ is at: 
   i) $\Sigma_A = \{0, 0, 0, 0, 0, 1\}$ for $F_6 = True$, or
   ii) $\Sigma_A = \{0, 0, 0, 0, 1, 0\}$ otherwise.

5. When $D$ plays $h_5$: $\max_j Z^{(5;j)}_A$ is at: 
   i) $\Sigma_A = \{0, 0, 0, 0, 0, 1\}$ for $F_3 = True$, or
   ii) $\Sigma_A = \{0, 0, 0, 1, 0, 0\}$ for $F_9 = True$ and $F_3 = False$, or
   iii) $\Sigma_A = \{0, 0, 1, 0, 0, 0\}$ otherwise, where $F_9$ is True if $I_A < \frac{(Cm_3 - Cm_5)/Cm_1}{p_5^{(3)} - p_5^{(5)}}$.

Different relations between $I_A$ and $T_A$, as expressed by conditions $F_1$ to $F_9$, create twelve separate regions in A’s problem, where the phrase ”region in A’s problem” is denoted by $RA$. In each $RA$, attacker $A$ has a different best response to $D$’s strategy profiles as shown in Table 4.4 and Figure 4.3.

Note that, from Figure 4.3, the attack strategy $m_1$ is dominated irrespective of the defender’s choices when the PU is not using the channel. The reason is the rationality of the attacker in selecting her actions considering PU activities and the existence of other attack strategies that perform at least as good as $m_1$.

Then, $D$’s SE strategies are calculated for each $RA$. To simplify the analysis, let $q_1, q_2, q_3 > \{q_2, q_3\}$ which means the time required to sense the channel $t_1 = Ck_1$ is greater than

\footnote{A particular player’s strategy is dominated when there exists another strategy that performs at least as good as the dominated strategy.}
Table 4.4: Regions in $A$’s problem when the PU not is using the channel

<table>
<thead>
<tr>
<th>RA</th>
<th>Condition(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$F_3 = True$</td>
</tr>
<tr>
<td>2</td>
<td>$F_3 = False$ and $F_1 = True$</td>
</tr>
<tr>
<td>3</td>
<td>$F_1 = False$, $F_5 = True$ and $F_6 = True$</td>
</tr>
<tr>
<td>4</td>
<td>$F_6 = False$, $F_7 = True$ and $F_9 = True$</td>
</tr>
<tr>
<td>5</td>
<td>$F_6 = False$, $F_7 = False$ and $F_9 = True$</td>
</tr>
<tr>
<td>6</td>
<td>$F_1 = False$, $F_5 = False$, $F_6 = True$ and $F_9 = True$</td>
</tr>
<tr>
<td>7</td>
<td>$F_1 = True$, $F_5 = False$ and $F_9 = True$</td>
</tr>
<tr>
<td>8</td>
<td>$F_7 = True$ and $F_9 = False$</td>
</tr>
<tr>
<td>9</td>
<td>$F_6 = False$, $F_7 = False$, $F_8 = True$ and $F_9 = False$</td>
</tr>
<tr>
<td>10</td>
<td>$F_1 = False$, $F_6 = True$ and $F_9 = False$</td>
</tr>
<tr>
<td>11</td>
<td>$F_1 = True$ and $F_9 = False$</td>
</tr>
<tr>
<td>12</td>
<td>$F_8 = False$</td>
</tr>
</tbody>
</table>

the time needed to transmit the sensing reports $t_2 = Ck_2$ or the time needed to transmit the spectrum decision $t_3 = Ck_3$.

Thus, $D$’s problem can be solved for each $RA$ as follows:

1. In $RA_1$:

$$\max_i (\sigma_{h_1}Z_D^{(1,6)} + \sigma_{h_2}Z_D^{(2,6)} + \sigma_{h_3}Z_D^{(3,6)} + \sigma_{h_4}Z_D^{(4,6)} + \sigma_{h_5}Z_D^{(5,6)}) \quad (4.18)$$

where $i$ is the index of $D$’s pure strategy, $Z_D^{(i,j)} = \Omega_D^{(i,j)}/Ck$. So, $D$’s best strategy against $A$ who is not willing to attack is not to defend, i.e. $D$ selects $h_5$.

16This assumption is very mild in nature as the QPs in the IEEE 802.22 standard can extend over one super-frame (approx. 160 ms) [2].
Thus, $D$’s maximum payoff is achieved by setting $\sigma_{h_i} = 1$ for the $i^{th}$ coefficient that holds the highest value of $Z_{D}^{(i;j)}$, where $I_D = U_D/Ck$. Accordingly, $D$’s maximum payoff is at: i) $\Sigma_D = \{0, 0, 0, 0, 1\}$ for $L_1 = True$, $Z_D^{(5;5)} = -p_s^{(5)} I_D$, or ii) at $\Sigma_D = \{0, 0, 1, 0, 0\}$ otherwise, $Z_D^{(3;6)} = -Ch_3/Ck$, where $L_1 = True$ if $I_D < \frac{Ch_3/Ck}{p_s^{(5)}}$.

3. Similarly, in RA$_3$: $D$’s maximum payoff is at: i) $\Sigma_D = \{0, 0, 0, 0, 1\}$ for $L_1 = True$, where $Z_D^{(5;5)} = -p_s^{(5)} I_D$, or ii) $\Sigma_D = \{0, 0, 1, 0, 0\}$ for $L_1 = False$ and $L_2 = True$, $Z_D^{(3;6)} = -Ch_3/Ck$, or iii) $\Sigma_D = \{1, 0, 0, 0, 0\}$ otherwise, $Z_D^{(1;5)} = p_s^{(1)} I_D - (1 - p_s^{(1)}) p_s^{(5)} I_D - (Ch_1/Ck)$, where $L_2 = True$ if $I_D < \frac{(Ch_1-Ch_3)/Ck}{p_s^{(1)}-p_s^{(5)}(1-p_s^{(1)})}$.

4. In RA$_4$: $D$’s maximum payoff is at: i) $\Sigma_D = \{0, 0, 0, 0, 1\}$ for $L_3 = True$, $Z_D^{(5;5)} = -p_s^{(5)} I_D$, or ii) $\Sigma_D = \{0, 0, 1, 0, 0\}$ otherwise, $Z_D^{(3;6)} = -Ch_3/Ck$, where $L_3 = True$ if $I_D < \frac{Ch_3/Ck}{p_s^{(1)}(1+p_s^{(5)})}$.
5. In $RA_5$: $D$’s maximum payoff is at: i) $\Sigma_D = \{0, 0, 0, 0, 1\}$ for $L_4 = True$, $Z_D^{(5,5)} = -p_s(5)I_D$, or ii) $\Sigma_D = \{0, 0, 0, 1, 0\}$ otherwise, $Z_D^{(4,5)} = p_\phi(1)I_D - (1 - p_\phi(1))p_s(5)I_D - (Ch_4/Ck)$, where $L_4 = True$ if $I_D < \frac{Ch_4/Ck}{p_\phi(1) - 1 - p_\phi(5)}$.

6. In $RA_6$: $D$’s maximum payoff is at: i) $\Sigma_D = \{0, 0, 0, 0, 1\}$ for $L_5 = True$, $Z_D^{(5,5)} = -p_s(5)I_D$, or ii) $\Sigma_D = \{1, 0, 0, 0, 0\}$ otherwise, $Z_D^{(1,5)} = p_\phi(1)I_D - (1 - p_\phi(1))p_s(5)I_D - Ch_1/Ck$, where $L_5 = True$ if $I_D < \frac{Ch_1/Ck}{p_\phi(1) - 1 - p_\phi(5)}$.

7. In $RA_7$: $D$’s maximum payoff is at: i) $\Sigma_D = \{0, 0, 0, 0, 1\}$ for $L_6 = True$, $Z_D^{(5,5)} = -p_s(5)I_D$, or ii) $\Sigma_D = \{0, 0, 0, 1, 0\}$ otherwise, $Z_D^{(4,6)} = -Ch_4/Ck$, where $L_6 = True$ if $I_D < \frac{Ch_4/Ck}{p_\phi(5)}$.

8. In $RA_8$: $D$’s maximum payoff is at: i) $\Sigma_D = \{0, 0, 0, 0, 1\}$ for $L_7 = True$, $Z_D^{(5,3)} = -p_s(3)I_D$, or ii) $\Sigma_D = \{0, 0, 1, 0, 0\}$ otherwise, $Z_D^{(3,5)} = p_\phi(1)I_D - (1 - p_\phi(1))p_s(5)I_D - Ch_3/Ck$, where $L_7 = True$ if $I_D < \frac{Ch_3/Ck}{p_\phi(1) + (1 + p_\phi(5)) + (p_\phi(4) - p_\phi(5))}$.

9. In $RA_9$: $D$’s maximum payoff is at: i) $\Sigma_D = \{0, 0, 0, 0, 1\}$ for $L_8 = True$, $Z_D^{(5,3)} = -p_s(3)I_D$, or ii) $\Sigma_D = \{0, 0, 0, 1, 0\}$ otherwise, $Z_D^{(4,5)} = p_\phi(1)I_D - (1 - p_\phi(1))p_s(5)I_D - Ch_4/Ck$, where $L_8 = True$ if $I_D < \frac{Ch_4/Ck}{p_\phi(1) - (1 + p_\phi(4)) - p_\phi(5)}$.

10. In $RA_{10}$: $D$’s maximum payoff is at: i) $\Sigma_D = \{0, 0, 0, 0, 1\}$ for $L_9 = True$, where $Z_D^{(5,3)} = -p_s(3)I_D$, or ii) $\Sigma_D = \{0, 0, 1, 0, 0\}$ for $L_9 = False$ and $L_{10} = True$, $Z_D^{(3,4)} = -p_s(4)I_D - Ch_3/Ck$, or iii) $\Sigma_D = \{1, 0, 0, 0, 0\}$ for otherwise, $Z_D^{(1,5)} = p_\phi(1)I_D - (1 - p_\phi(1))p_s(5)I_D - Ch_1/Ck$, where $L_9 = True$ if $I_D < \frac{Ch_3/Ck}{p_\phi(5) - p_\phi(4)}$, $L_{10} = True$ if $I_D < \frac{(Ch_3 - Ch_4) / Ck}{p_\phi(1) - (1 + p_\phi(4)) - p_\phi(5)}$.

11. In $RA_{11}$: $D$’s maximum payoff is at: i) $\Sigma_D = \{0, 0, 0, 0, 1\}$ for $L_9 = True$, where $Z_D^{(5,3)} = -p_s(3)I_D$, or ii) $\Sigma_D = \{0, 0, 1, 0, 0\}$ for $L_9 = False$ and $L_{11} = True$, $Z_D^{(3,4)} = -p_s(4)I_D - Ch_3/Ck$, or iii) $\Sigma_D = \{0, 0, 0, 1, 0\}$ for otherwise, $Z_D^{(4,6)} = -Ch_4/Ck$, where $L_{11} = True$ if $I_D < \frac{Ch_4/Ck}{p_\phi(5)}$. 

62
12. In RA12: D’s maximum payoff is at: i) $\Sigma_D = \{0, 0, 0, 0, 1\}$ for $L_{12} = True$, where $Z_D^{(5;3)} = -p_s^{(3)} I_D$, or ii) $\Sigma_D = \{0, 0, 1, 0, 0\}$ for $L_{12} = False$ and $L_{10} = True$, $Z_D^{(3;3)} = p_\phi^{(1)} I_D - (1 - p_\phi^{(1)}) p_s^{(3)} I_D - Ch_3/Ck$, or iii) $\Sigma_D = \{0, 0, 0, 1, 0\}$ for otherwise, $Z_D^{(4;5)} = -Ch_4/Ck$, where $L_{12} = True$ if $I_D < \frac{(Ch_4-Ch_3)/Ck}{(1-p_\phi^{(1)})(p_s^{(3)}-p_s^{(5)})}$.

In conclusion, different values of $I_D$ (expressed by conditions $L_1$ to $L_{12}$) and $I_A$ and $T_A$ (represented by conditions $F_1$ to $F_9$) create 27 regions of equilibria $R_{SE}$ in game $G$ when the PU is not using the channel as shown in Figure 4.4.

It is clear that $R_{SE} = 1$, in Figure 4.4, is the No-Attack/No-defense region in which $A/D$ will not attack/defend the channel. Moreover, in areas $R_{SE} = \{3, 5, 8, 14, 24\}$, the attacker prefers not to attack (expressed by attack strategy $m_6$) due to the deployed honeypots. Thus, player $D$ can reduce the probability of success of the deceiving attack to almost 0% (in the regions above) by utilizing deception.

4.3.2.2 When the PU is using the channel

Similarly, A’s best responses are calculated using Table 4.2 and Eqns. (4.7) and (4.8). Therefore, different relations between $I_A$ and $T_A$ create eleven regions in $A$’s problem when
Table 4.5: Regions in A’s problem when the PU is using the channel

<table>
<thead>
<tr>
<th>RA’</th>
<th>Condition(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$F'_2 = True$</td>
</tr>
<tr>
<td>2</td>
<td>$F'_1 = True, F'_3 = True$ and $F_1 = True$</td>
</tr>
<tr>
<td>3</td>
<td>$F'_1 = False, F'_4 = True$ and $F'_7 = True$</td>
</tr>
<tr>
<td>4</td>
<td>$F'_1 = False, F'_1 = False$ and $F'_7 = True$</td>
</tr>
<tr>
<td>5</td>
<td>$F'_1 = True, F'_3 = False$ and $F'_7 = True$</td>
</tr>
<tr>
<td>6</td>
<td>$F'_4 = True, F'_5 = True$ and $F'_7 = False$</td>
</tr>
<tr>
<td>7</td>
<td>$F'_1 = False, F'_4 = False, F'_5 = True$ and $F'_7 = False$</td>
</tr>
<tr>
<td>8</td>
<td>$F'_1 = True, F'_5 = True$ and $F'_7 = False$</td>
</tr>
<tr>
<td>9</td>
<td>$F'_5 = False$ and $F'_6 = False$</td>
</tr>
<tr>
<td>10</td>
<td>$F'_1 = False, F'_5 = False$ and $F'_6 = True$</td>
</tr>
<tr>
<td>11</td>
<td>$F'_1 = True$ and $F'_5 = False$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$F'_1$</th>
<th>$I_A &lt; \frac{p_1^{(1)}TA + Cm_5/Cl}{p_5^{(5)}(1-p_6^{(1)})}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$F'_2$</td>
<td>$I_A &lt; \frac{Cm_5/Cl}{p_6^{(5)}}$</td>
</tr>
<tr>
<td>$F'_3$</td>
<td>$I_A &lt; \frac{Cm_2/Cl}{p_4^{(2)}}$</td>
</tr>
<tr>
<td>$F'_4$</td>
<td>$I_A &lt; \frac{-p_1^{(1)}TA + (Cm_2-Cm_5)/Cl}{p_4^{(2)} - (1-p_6^{(1)}p_5^{(5)})}$</td>
</tr>
<tr>
<td>$F'_5$</td>
<td>$I_A &lt; \frac{(Cm_1-Cm_5)/Cl}{(1-p_6^{(1)})(p_4^{(1)} - p_5^{(5)})}$</td>
</tr>
<tr>
<td>$F'_6$</td>
<td>$I_A &lt; \frac{p_1^{(1)}TA + (Cm_1-Cm_2)/Cl}{p_4^{(1)}(1-p_6^{(1)} - p_6^{(2)})}$</td>
</tr>
<tr>
<td>$F'_7$</td>
<td>$I_A &lt; \frac{(Cm_1-Cm_5)/Cl}{(p_4^{(1)} - p_5^{(5)})}$</td>
</tr>
</tbody>
</table>

the PU is using the channel, denoted by $(RA')$, as shown in Table 4.5. Solving $D$’s problem in each $RA'$ in Table 4.5 for different values of $I_D$ yields a set of conditions expressed by $L'_1$ to $L'_11$ in Table 4.6.

In conclusion, various conditions on $I_D, I_A$ and $T_A$ create 25 regions of equilibrium $R'_SE$ in game $G$ when the PU is using the channel as shown in Figure 4.5. Clearly, the region $R'_{SE} = 1$, in Figure 4.5, is the No-Attack/No-defense region and in regions $R_{SE} = \{3, 10, 25\}$, player $D$ can reduce the probability of success of the deceiving attack to almost 0% by utilizing deception.

Three main observations from the results. First, the attack strategies $m_1$ and $m_2$ are
Table 4.6: Summary of conditions in D’s problem when the PU is using the channel

<table>
<thead>
<tr>
<th>Index</th>
<th>Condition</th>
<th>Index</th>
<th>Condition</th>
<th>Index</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L'_1$</td>
<td>$I_D &lt; \frac{C_{h3}/Cl}{p_s^{(5)}}$</td>
<td>$L'_2$</td>
<td>$I_D &lt; \frac{C_{h3}/C_k}{p_s^{(2)}(1+p_s^{(5)})}$</td>
<td>$L'_3$</td>
<td>$I_D &lt; \frac{C_{h4}/C_k}{p_o^{(1)}(1+p_o^{(5)})}$</td>
</tr>
<tr>
<td>$L'_4$</td>
<td>$I_D &lt; \frac{C_{h3}/C_k}{(p_s^{(3)}p_o^{(5)}+p_s^{(1)})}$</td>
<td>$L'_5$</td>
<td>$I_D &lt; \frac{(C_{h4}-C_{h3}/C_k}{p_s^{(2)}}$</td>
<td>$L'_6$</td>
<td>$I_D &lt; \frac{C_{h4}/C_k}{p_o^{(3)}(1-p_o^{(3)})p_s^{(5)}+p_s^{(1)}}$</td>
</tr>
<tr>
<td>$L'_7$</td>
<td>$I_D &lt; \frac{C_{h4}/C_k}{p_o^{(1)}(1-p_o^{(3)})p_s^{(5)}+p_s^{(1)}}$</td>
<td>$L'_8$</td>
<td>$I_D &lt; \frac{C_{h3}/C_k}{p_s^{(2)}(1+p_s^{(5)})}$</td>
<td>$L'_9$</td>
<td>$I_D &lt; \frac{(C_{h4}-C_{h3}/C_k}{p_s^{(2)}$</td>
</tr>
<tr>
<td>$L'_10$</td>
<td>$I_D &lt; \frac{C_{h3}/C_k}{p_s^{(3)}(1+p_s^{(5)})}$</td>
<td>$L'_11$</td>
<td>$I_D &lt; \frac{(C_{h4}-C_{h3}/C_k}{(1-p_s^{(3)})p_s^{(4)}-p_s^{(5)}$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 4.5: SE Regions in $G$ when the PU is using the channel

*Dominated* when the PU is *not using* the channel irrespective of $\Sigma_D$ or $I_A$. Similarly, $m_3$ and $m_4$ are *dominated* when the PU is *using* the channel. This observation confirms the rationality of the attacker regarding the PU activity, where $A$ chooses the attack strategy that induces the highest impact on the victim CRNs by considering the PU activity over the channel.

Second, $I_A = \frac{C_{m_3}/Cl}{p_s^{(5)}}$ represents the attack threshold, below which $A$ will not launch any attacks. This observation confirms the common belief about $A$’s behavior when expecting a high cost to gain ratio from attacking a channel. The importance of this observation is in defining the exact attack threshold.
Finally, \( \frac{\text{Ch}_3/Ck}{p_5^{(1)}+(1-p_5^{(3)})p_5^{(5)}+p_5^{(3)}} \) and \( \frac{\text{Ch}_3/Ck}{p_5^{(1)}-(1-p_5^{(3)})p_5^{(5)}+p_5^{(3)}} \) represent the defense threshold when the PU is not using the channel and using the channel, respectively. Thus, if \( I_D \) is lower than the defense threshold, \( D \) does not defend the channel. Small \( I_D \) occurs when the security cost to gain ratio is high. Thus, \( D \) would prefer to switch to another free channel rather than defending the current channel.

### 4.3.3 The General Case: A Game with Incomplete Information

In the general case of game \( G \), players are uncertain about PU activities, yielding a Bayesian Stackelberg game. Besides, the defender can play mixed strategy profiles, meaning \( \sigma_{hi} \in [0, 1] \). Let \( \alpha \) be the probability that the PU is not using the channel, so \( (1-\alpha) \) is the probability that the PU is using the channel. The probability distribution over the PU activity pattern is assumed to be common knowledge in the game. For example, the players can obtain the probability distribution over the PU activity pattern through learning as proposed in [98].

With the target to solve the Bayesian game, the Harsanyi transformation is applied that converts the incomplete information game into a complete information game effectively [71]. Thus, existing methods can be used to find the SE strategies in a complete information game settings [97, 99].

To this end, let

\[
\beta_D^{\Sigma} = \sum_{i=1}^{\mid \mathcal{H} \mid} \sum_{j=1}^{\mid \mathcal{M} \mid} \sigma_{hi} \sigma_{mj} (\alpha Z_{D1}^{(ij)} + (1-\alpha)Z_{D2}^{(ij)}) \quad (4.20a)
\]

\[
\beta_A^{\Sigma} = \sum_{i=1}^{\mid \mathcal{H} \mid} \sum_{j=1}^{\mid \mathcal{M} \mid} \sigma_{hi} \sigma_{mj} (\alpha Z_{A1}^{(ij)} + (1-\alpha)Z_{A2}^{(ij)}) \quad (4.20b)
\]

where \( Z_{D1}^{(ij)} \) and \( Z_{D2}^{(ij)} \) are \( D \)'s normalized payoff when the PU is not using the channel and using the channel respectively. Similarly, \( Z_{A1}^{(ij)} \) and \( Z_{A2}^{(ij)} \) are \( A \)'s payoff when the PU is not using the channel and using the channel, respectively. Then, \( D \)'s problem can be mathematically expressed in \( \mathcal{P}3 \) as:
\[ \mathcal{P}_3 : \]
\[
\max_{\Sigma_D} \beta^{\Sigma_D, g(\Sigma_D)}_D \\
\text{subject to:}
\]
\[
\forall \Sigma_D, g' : \beta^{\Sigma_D, g(\Sigma_D)}_A \geq \beta^{\Sigma_D, g'(\Sigma_D)}_A \quad (4.21b)
\]
\[
\forall \Sigma_D : \beta^{\Sigma_D, g(\Sigma_D)}_D \geq \beta^{\Sigma_D, \eta(\Sigma_D)}_D \quad (4.21c)
\]
\[
\sigma_{m_j} \in \{0, 1\}, \quad \sum_{j=1}^{\mid M \mid} \sigma_{m_j} = 1 \quad (4.21d)
\]
\[
\sigma_{h_i} \in [0, 1], \quad \sum_{i=1}^{\mid H \mid} \sigma_{h_i} = 1 \quad (4.21e)
\]
\[
\alpha \in [0, 1] \quad (4.21f)
\]

Due to the nonlinearity in (4.21), solving \( \mathcal{P}_3 \) to find the SE in closed form is rather hard. Instead, a Matlab-based simulation is conducted to find the SE of the general form of game \( \mathcal{G} \).

4.4 The IEEE 802.22 Nash Deception-based Game Problem

For comparison, the points of Nash equilibria (NE) in game \( \mathcal{G} \) are calculated in this section. In NE model, the game structure and the payoff functions are assumed to be known to the players. In addition, \( A \) cannot observe \( D \)'s deployed defense strategies before making her decision, so the players move simultaneously.

A pair of strategy profiles \( (\Sigma^*_A, \Sigma^*_D) \) forms a NE if the pair satisfies the following conditions:

\[
\forall \Sigma_A : \Omega^{\Sigma_A, \Sigma^*_A}_A \geq \Omega^{\Sigma_A, \Sigma_A}_A \quad (4.22a)
\]
\[
\forall \Sigma_D : \Omega^{\Sigma_D, \Sigma^*_A}_D \geq \Omega^{\Sigma_D, \Sigma^*_A}_D \quad (4.22b)
\]

where (4.22a) and (4.22b) represent \( A \) and \( D \) playing their best responses, respectively.
At any of the points of NE, no player is getting a higher payoff from solely changing her selected strategy. Finding the points of NE in game problem $G$ aims at calculating the decision variables $\sigma_m \forall j$ and $\sigma_h \forall i$.

In the sequel, the NE is calculated when the PU is not using the frequency channel and when the PU is using the frequency channel as shown in Table 4.1 and Table 4.2, respectively. First, the special case is solved, where the players commit to pure strategies, i.e $\sigma_m \in \{0, 1\}, \sum_{j=1}^{6} \sigma_m = 1$ and $\sigma_h \in \{0, 1\}, \sum_{i=1}^{5} \sigma_h = 1$. Then, the general case is solved where the players play mixed strategies, i.e $\sigma_m \in [0, 1], \sum_{j=1}^{6} \sigma_m = 1$ and $\sigma_h \in [0, 1], \sum_{i=1}^{5} \sigma_h = 1$.

4.4.1 The special case: pure strategy NE

Using Table 4.1, $A$’s normalized payoffs $(Z^{(j)}_A)$ from playing each pure attack strategy $m_j$ over all pure defense strategies $h_i$ when the PU is not using the channel are:
\[
\begin{cases}
p_s^{(5)} I_A \left( \sigma_{h_1} (1 - p_\phi^{(3)}) + \sigma_{h_2} (1 - p_\phi^{(1)}) + \sigma_{h_3} (1 - p_\phi^{(1)}) + \sigma_{h_4} (1 - p_\phi^{(2)}) + \sigma_{h_5} \right) \\
-T_A \left( \sigma_{h_1} p_\phi^{(3)} + \sigma_{h_2} p_\phi^{(1)} + \sigma_{h_3} p_\phi^{(1)} + \sigma_{h_4} p_\phi^{(2)} \right) - C m_1 / C l, & j = 1 \\
-T_A \left( \sigma_{h_1} p_\phi^{(3)} + \sigma_{h_2} p_\phi^{(1)} + \sigma_{h_4} p_\phi^{(1)} \right) - C m_2 / C l, & j = 2 \\
p_s^{(3)} I_A \left( \sigma_{h_1} (1 - p_\phi^{(3)}) + \sigma_{h_2} (1 - p_\phi^{(1)}) + \sigma_{h_3} (1 - p_\phi^{(1)}) + \sigma_{h_4} (1 - p_\phi^{(2)}) + \sigma_{h_5} \right) \\
-T_A \left( \sigma_{h_1} p_\phi^{(3)} + \sigma_{h_2} p_\phi^{(1)} + \sigma_{h_3} p_\phi^{(1)} + \sigma_{h_4} p_\phi^{(2)} \right) - C m_3 / C l, & j = 3 \\
p_s^{(4)} I_A \left( \sigma_{h_1} (1 - p_\phi^{(2)}) + \sigma_{h_2} (1 - p_\phi^{(1)}) + \sigma_{h_3} + \sigma_{h_4} (1 - p_\phi^{(1)}) + \sigma_{h_5} \right) \\
-T_A \left( \sigma_{h_1} p_\phi^{(2)} + \sigma_{h_2} p_\phi^{(1)} + \sigma_{h_4} p_\phi^{(1)} \right) - C m_4 / C l, & j = 4 \\
p_s^{(5)} I_A \left( \sigma_{h_1} (1 - p_\phi^{(1)}) + \sigma_{h_2} + \sigma_{h_3} (1 - p_\phi^{(1)}) + \sigma_{h_4} (1 - p_\phi^{(1)}) + \sigma_{h_5} \right) \\
-T_A \left( \sigma_{h_1} p_\phi^{(1)} + \sigma_{h_3} p_\phi^{(1)} + \sigma_{h_4} p_\phi^{(1)} \right) - C m_5 / C l, & j = 5 \\
0, & j = 6 
\end{cases}
\]

where \( Z_A^{(j)} = \Omega_A^{(j)} / C l \), \( T_A = \phi / C l \) and \( I_A = U_A / C l \) representing \( A \)'s normalized payoff function, \( A \)'s deterrent factor and \( A \)'s incentive factor, respectively. Similarly, \( D \)'s normalized
payoffs \((Z_D^{(i)})\) from playing each defense strategy are:

\[
\left\{
\begin{array}{l}
I_D(\sigma_{m_1}p_\phi^{(5)} + \sigma_{m_2}p_\phi^{(2)} + \sigma_{m_3}p_\phi^{(3)} + \sigma_{m_4}p_\phi^{(2)} + \sigma_{m_5}p_\phi^{(1)} - \sigma_{m_1}p_s^{(1)}(1 - p_\phi^{(3)}) - \sigma_{m_3}p_s^{(1)}(1 - p_\phi^{(3)}) - \sigma_{m_4}p_s^{(1)}(1 - p_\phi^{(3)}) - \sigma_{m_5}p_s^{(1)}(1 - p_\phi^{(3)})), \quad i = 1 \\
I_D(\sigma_{m_1}p_\phi^{(1)} + \sigma_{m_2}p_\phi^{(1)} + \sigma_{m_3}p_\phi^{(1)} + \sigma_{m_4}p_\phi^{(1)} - \sigma_{m_1}p_s^{(5)}(1 - p_\phi^{(1)}) - \sigma_{m_3}p_s^{(5)}(1 - p_\phi^{(1)})) - Ch_2/Ck, \quad i = 2 \\
I_D(\sigma_{m_1}p_\phi^{(1)} + \sigma_{m_3}p_\phi^{(1)} + \sigma_{m_5}p_\phi^{(1)} - \sigma_{m_1}p_s^{(5)}(1 - p_\phi^{(1)}) - \sigma_{m_3}p_s^{(5)}(1 - p_\phi^{(1)})) - Ch_3/Ck, \quad i = 3 \\
I_D(\sigma_{m_1}p_\phi^{(2)} + \sigma_{m_2}p_\phi^{(1)} + \sigma_{m_3}p_\phi^{(2)} + \sigma_{m_4}p_\phi^{(1)} + \sigma_{m_5}p_\phi^{(1)} - \sigma_{m_1}p_s^{(5)}(1 - p_\phi^{(2)})) - Ch_4/Ck, \quad i = 4 \\
-I_D(\sigma_{m_1}p_s^{(5)} + \sigma_{m_3}p_s^{(3)} + \sigma_{m_4}p_s^{(4)} + \sigma_{m_5}p_s^{(5)}), \quad i = 5 \\
\end{array}
\right.
\]

where \(Z_D^{(i)} = \Omega_D^{(i)} / Ck\) and \(I_D = U_D / Ck\).

From (4.23) and (4.24), the following results are obtained:

1. If \(D\)'s best strategy is not to defend, i.e. \(\Sigma_D^* = \{0, 0, 0, 0, 1\}\), then \(A\)'s best strategy is not to attack, \(\Sigma_A^* = \{0, 0, 0, 0, 1\}\) if \(I_A < \frac{C_{m_5}/C_l}{p_s^{(5)}}\), irrespective of the deterrent factor \(T_A\).

2. Irrespective of \(I_A\) and \(T_A\), if \(I_D < \frac{Ch_3/Ck}{p_\phi^{(1)}(1 + p_s^{(3)})}\), then \(D\)'s best strategy is not to defend, \(\Sigma_D^* = \{0, 0, 0, 0, 1\}\). In this case, \(A\)'s best strategies are calculated as follows:

\[
Z_A^{h_5} = \sigma_{m_1}(p_s^{(5)}) I_A - Cm_1/C_l + \sigma_{m_2}(-Cm_2/C_l) + \sigma_{m_3}(p_s^{(3)}) I_A - Cm_3/C_l + \sigma_{m_4}(p_s^{(4)}) I_A - Cm_4/C_l + \sigma_{m_5}(p_s^{(5)}) I_A - Cm_5/C_l + \sigma_{m_6}(0) \quad (4.25)
\]

70
So, A’s best strategy is calculated by assigning unity to the jth strategy that maximizes (4.25). Thus, A’s best strategy merely depends on IA: i) \( \Sigma_A^* = \{0, 0, 0, 0, 1\} \) for \( I_A < \frac{(Cm_5/Cl)/p_s^{(5)}}{p_s^{(3)}-p_s^{(5)}} \), \( Z_A^{h_5;m_6} = 0 \), or ii) \( \Sigma_A^* = \{0, 0, 0, 0, 0\} \) for \( (Cm_5/Cl)/p_s^{(5)} \leq I_A < \frac{(Cm_4-Cm_5)/Cl}{p_s^{(4)}-p_s^{(5)}} \), \( Z_A^{h_5;m_5} = p_s^{(5)}U_A - Cm_5/Cl \), or iii) \( \Sigma_A^* = \{0, 0, 1, 0, 0\} \) for \( I_A \geq \frac{(Cm_3-Cm_5)/Cl}{p_s^{(3)}-p_s^{(5)}} \), \( Z_A^{h_5;m_3} = p_s^{(3)}U_A - Cm_3/Cl \).

If \( I_D \geq \frac{Ch_3/Ck}{p_\phi^{(3)}(1+p_s^{(3)})} \), and \( I_A \geq \frac{Cm_5/Cl}{p_s^{(5)}} \), i.e., both A and D have the incentive to attack and defend the channel, respectively, then, the NE is calculated by finding the intersection between player’s best responses. Consequently, from (4.23), A’s best responses are:

\[
\begin{align*}
Z_A^{m_1} &= p_s^{(5)}I_A - Cm_1, \quad i \in \{5\} \\
Z_A^{m_2} &= -Cm_2, \quad i \in \{3, 5\} \\
Z_A^{m_3} &= p_s^{(3)}I_A - Cm_3, \quad i = 5 \\
Z_A^{m_4} &= p_s^{(4)}I_A - Cm_4, \quad i \in \{3, 5\} \\
Z_A^{m_5} &= p_s^{(5)}I_A - Cm_5, \quad i \in \{2, 5\} \\
Z_A^{m_6} &= 0 \quad \forall i
\end{align*}
\]

(4.26)

It is clear from (4.26) that, regardless of D’s deception strategy, \( m_2 \) is a dominated strategy for the attacker. The reason is because A always has another attack strategy that performs at least as good as \( m_2 \). Thus, \( m_2 \) is not considered in A’s best responses.

Moreover, from (4.24), D’s best responses are as follows:

\[
\begin{align*}
Z_D^{h_1} &= p_\phi^{(3)}U_D - (1-p_\phi^{(3)})p_s^{(3)}U_D - Ch_1, \quad j \in \{3\} \\
Z_D^{h_2} &= p_\phi^{(1)}U_D - Ch_2, \quad j = 2 \\
Z_D^{h_3} &= p_\phi^{(1)}U_D - (1-p_\phi^{(1)})p_s^{(5)}U_D - Ch_3, \quad j \in \{1, 5\} \\
Z_D^{h_4} &= p_\phi^{(2)}U_D - (1-p_\phi^{(2)})p_s^{(5)}U_D - Ch_4, \quad j = 3 \\
Z_D^{h_5} &= 0, \quad j \in \{2, 6\}
\end{align*}
\]

(4.27)

From (4.26) and (4.27), no possible intersection between A’s best responses and D’s best
responses. Thus, no pure strategy NE exists when \( I_D \geq \frac{C_{h_5}/C_k}{p_s^{(1)}(1+p_s^{(3)})} \), and \( I_A \geq \frac{C_{m_5}/C_l}{p_s^{(5)}} \).

Similarly, when the PU is using the channel, Table 4.2 is used to calculate \( A \)'s/\( D \)'s payoffs from playing each pure attack/defense strategies over all pure defense/attack strategies. The following results are obtained.

1. If \( D \)'s best strategy is not to defend, i.e. \( \Sigma_D^* = \{0, 0, 0, 0, 1\} \), then \( A \)'s best strategy is not to attack, \( \Sigma_A^* = \{0, 0, 0, 0, 1\} \) if \( I_A < \frac{C_{m_5}/C_l}{p_s^{(5)}} \), irrespective of the deterrent factor \( T_A \).

2. Irrespective of \( I_A \) and \( T_A \), if \( I_D < \frac{C_{h_5}/C_k}{p_s^{(1)}(1+p_s^{(3)})} \), then \( D \)'s best strategy is not to defend, \( \Sigma_D^* = \{0, 0, 0, 0, 1\} \), so \( A \)'s payoff is as follows:

\[
Z_A^{h_5} = \sigma_{m_1}(p_s^{(1)}I_A - Cm_1/C_l) + \sigma_{m_2}(p_s^{(2)}I_A - Cm_2/C_l) + \sigma_{m_3}(p_s^{(1)}I_A - Cm_3/C_l)
+ \sigma_{m_4}(p_s^{(2)}I_A - Cm_4/C_l) + \sigma_{m_5}(p_s^{(5)}I_A - Cm_5/C_l) + \sigma_{m_6}(0)
\]  

(4.28)

Similarly, \( A \)'s best strategy is calculated by assigning unity to the \( j \)th strategy that maximizes (4.28). Thus, \( \Sigma_A^* \) merely depends on \( I_A \) such that: i) \( \Sigma_A^* = \{0, 0, 0, 0, 1\} \) for \( I_A < \frac{C_{m_5}/C_l}{p_s^{(5)}} \), \( Z_A^{h_5;m_6} = 0 \), or ii) \( \Sigma_A^* = \{0, 0, 0, 0, 1, 0\} \) for \( \frac{C_{m_5}/C_l}{p_s^{(5)}} \leq I_A < \frac{(C_{m_1}-C_{m_5})/C_l}{p_s^{(3)}-p_s^{(5)}} \), \( Z_A^{h_5;m_5} = p_s^{(5)}I_A - Cm_5/C_l \), or iii) \( \Sigma_A^* = \{1, 0, 0, 0, 0\} \) for \( I_A \geq \frac{(C_{m_1}-C_{m_5})/C_l}{p_s^{(3)}-p_s^{(5)}} \), \( Z_A^{h_5;m_1} = p_s^{(1)}I_A - Cm_1/C_l \).

3. If \( I_D \geq \frac{C_{h_5}/C_k}{p_s^{(1)}(1+p_s^{(3)})} \), and \( I_A \geq \frac{C_{m_5}/C_l}{p_s^{(5)}} \), i.e., both \( D \) and \( A \) have the incentive to defend and attack the channel, respectively, then, no pure strategy NE exists in the game.

The reason behind the non-existence of the pure strategy NE when both players have the incentive to engage in the game is that when \( A \) selects a strategy \( m_j \) that includes attacking one or more CR network’s vulnerabilities, \( D \) is better selecting a deception strategy \( h_i \) that includes protecting the selected vulnerabilities. In this case, it is better for \( A \) to select other pure strategies that include attacking different CR network vulnerabilities, hence, no pure strategy NE exists in the game.
4.4.2 The general case: mixed strategy NE

In non-cooperative, normal form games, such as the game problem \( \mathcal{G} \), there exists at least one mixed strategy that satisfies (4.22) \([100]\). In a practical manner, the mixed strategy NE is reached when \( D \) and \( A \) expect each other’s strategy with the associated probabilities and both \( D \) and \( A \) play the expected response. Hence, the equilibrium can be determined by searching for the possible combinations of the players’ strategies. However, finding the closed form of the mixed strategy NE in \( \mathcal{G} \) is rather hard because of the relatively large number of combinations of players’ pure strategies (e.g., 30 in this thesis = product of 5 defense and 6 attack strategies).

In the literature, many algorithms exist for solving the NE, for example, \([101, 102]\). In this thesis, the algorithm in \([101]\) is adopted because of its high efficiency in finding the NE in two player games \([24]\). The chosen algorithm is referred to as LH algorithm in Section 4.5.

4.5 Simulation Results and Interpretation

In this section, a \emph{Matlab}–based simulation of the game problem \( \mathcal{G} \) in (4.21) is conducted and compared to the Nash equilibrium of the same game which was calculated using the LH algorithm \([24]\). The simulation is made to demonstrate the usefulness of the proposed work in combating the deceiving attack in CRNs. Moreover, the simulation provides a comparison between the SE and the NE in game \( \mathcal{G} \).

The simulation setup is as follows:

1. \( I_A, I_D \) and \( T_A \) independently increase from 1 to 40.

2. The probability of success of attack strategies \( p_s^{(j)} \) and the probability of falling into honeypots \( p_\phi^{(i,j)} \) are as calculated in Section 4.3.1.2.

3. \( \alpha = 0.5 \), representing 50% chance that the PU is \emph{not using} the channel.
Figure 4.6: The players’ strategy profiles Σ for different values of deterrent factor $T_A$ in two game scenarios $Sc1$ and $Sc2$ representing a defender with low and high incentive $I_D$, respectively.

Figure 4.6 is formed of four graphs: the upper two charts show $D$’s deception strategy $\Sigma_D$ on the $y$-axis against different values of attacker’s incentive $I_A$ on the $x$-axis and the lower two are for $A$’s attack strategies against $I_A$. Similarly, Figure 4.7 is formed of four graphs: the upper two for normalized payoff ($Z_D$) against $I_A$ and the lower two are for $A$’s normalized payoff $Z_A$. Figs. 4.6 and 4.7 show the results under two game situations: $a)$ when $D$ cannot impose a high penalty on $A$, represented by $T_A = 1$ (figures on the left) and $b)$ when $D$ can impose a high penalty on $A$, represented by $T_A = 20$ (graphs on the right).

In situation $(a)$, the deployed honeypots hold the lowest effect on $A$ as she only loses the implementation cost of the launched attack actions which fell into honeypots.

The simulation results are better explained by introducing two defense scenarios. The first
defense scenario, Sc1, represents $D$ with a little incentive ($I_D = 1$). The low $I_D$ occurs if the defense budget is limited, or if the number of available free channels is high. Consequently, $D$ would prefer to switch to another open channel rather than defending the current channel. The second defense scenario, Sc2, represents $D$ with a high incentive $I_D = 20$.

In Figure 4.6, under Sc1, the defense is only deployed when higher $I_A$ is expected, irrespective of $T_A$. In particular, if $I_A$ is lower than the attack threshold ($\frac{C_{m_5}/C_l}{P_{a_5}} = 3.33$) no attacks are expected. Yet, if $I_A \geq 3.33$, $A$ uses $m_5$ to jam the spectrum decision to isolate the SUs within the jamming range attaining limited impact on the CRN with lower cost. Finally, if $I_A$ is very high ($I_A \geq 38$), $A$ uses attack strategy $m_3$ which includes emulation of the PU signal, jamming the spectrum reports and jamming the spectrum decision, representing the most aggressive attack attempt.

In defense scenario Sc2, $D$’s optimum strategy merely depends on $I_A$ and $T_A$ as follows: i) if $T_A$ is low (graphs on the left), the honeypots exert a little impact on $A$, thus $D$ tends to implement more honeypots, forcing $A$ to mostly not to attack. ii) if $T_A$ is high (graphs on the right), $D$ considerably implements fewer honeypots, yet holding the same effect on $A$ as the attacker $A$ is mostly not attacking the channel. The effect of $T_A$ is further illustrated by comparing the results in Figure 4.6(d) where $A$ does not attack if $I_A < 15$, to the results in Figure 4.6(c) where $A$ attacks when $I_A > 4$.

Figure 4.7 compares the players’ payoff from playing the NE to the SE in the game. It is apparent from the results that $D$ is better playing the Stackelberg model rather than the Nash model because $D$’s payoff from the SE is at least as high as the defender’s payoff from the NE irrespective of $I_D$, $I_A$ and $T_A$. The increased payoff for the leader in the Stackelberg model is commonly known as the commitment reward [61]. To ensure the leadership of $D$, she may announce the security structure of the proposed deception-based defense mechanism including the honeypots’ design and the honeypots’ deployment probabilities. $D$ should obscure only the exact schedule of the honeypots. In addition, region $R_l$ in Figure 4.7 shows
Figure 4.7: The players’ normalized payoff ($Z$) for different values of deterrent factor $T_A$ in two game scenarios $Sc_1$ and $Sc_2$ representing a defender with low and high incentive $I_D$, respectively.

the no-attack-no-defense region where the attacker incentive is below the attack threshold ($I_A = \frac{C_{\text{max}}/C_l}{p_s^{(5)}} = 3.3$), thus no attack (consequently no defense) is expected.

Most importantly, the usefulness of the proposed defense scheme is pointed out through comparing $Sc_1$, where no defense is deployed, to $Sc_2$ where partial or full protection is in place. In $Sc_1$, the attacker’s payoffs solely depend on the attacker aggressiveness. However, in $Sc_2$, the probability of attack success is decreased to nearly 0% (by forcing $A$ to use $m_6$ instead of $m_5$) or 30% (by forcing $A$ to use $m_5$ instead of $m_3$) as shown in Figs. 4.6(c) and 4.6(d).

Crucially, the accuracy of the proposed work is highly affected by the security assessment process that precedes the security planning process. Typically, it is required to guesstimate accurately: $i$) the attacker’s strategies and their associated probabilities of success, $ii$) the attacker’s payoff function and $iii$) the attacker’s incentive and deterrent factors.
4.6 Chapter Summary

In conclusion, both the analytical work and the numerical results proved the success of the deception strategies in combating the deceiving attack. A defender with high incentive to defend the channel can utilize the proposed deception-based defense mechanism to reduce the probability of success of the deceiving attack to nearly 0% irrespective of the PU activity over the targeted channel.

Contrary to popular belief, the defender is better in declaring the security structure of the defense mechanism to enforce the Stackelberg model which results in a higher payoff for the leader. Besides, an acute attacker is always observing the defense strategies before choosing her best attack strategy which also strengthens the utilization of the Stackelberg equilibria as a pragmatic approach in modeling the proposed game. Most importantly, the accuracy of the security assessment process that precedes the security planning process is vital in adjusting the defender’s selection of the deception strategies.
Chapter 5

Learning in Repeated CRN Security Games

5.1 Introduction

At first glance, the CRN appears to have an advantage over the jamming attackers because of its ability to change the operating frequencies to avoid interference. In practice, the advanced jamming attacker(s) might determine and follow the utilized frequency channels of the target CRN to re-engage. Thus, the interaction with the advanced jamming attackers takes place frequently on repeated intervals\textsuperscript{17} [15, 16].

Besides, the CRN’s spectrum sensing times can be considered as the arena for the interactions between the jamming attacker and the defending CRN. Broadly, the repetition rate of the spectrum sensing times in CRNs is very high, for instance, in [103] the optimum spectrum sensing time was estimated to be 6ms every 100ms of useful communication over the channel. Thus, the interactions between the jamming attacker(s) and the CRN might take place approximately 9 times every single second.

In a different context, in the preceding chapter, the game theory was utilized to describe/analyze the interactions between the jamming attacker(s) and the defending CRN in a single-run security problem $G$. Thus, introducing a solution (i.e., a particular deployment probability of deception strategies) that guarantees a certain payoff for the defender and the attacker under game equilibria. The calculated game-theoretic solution suffers from the following limitations:

\begin{itemize}
  \item 1. The solution’s sensitivity to the assumed attacker’s behavioral model. Put differently, the calculation of the points of Stackelberg equilibria in $G$ is based
\end{itemize}

\textsuperscript{17}In the literature of the security of wireless communications, this type of attacker is called the frequency-follower jammer and is designed to target the frequency-hopping-based networks.

78
upon a guesstimated information about the attacker’s preferences that might be inaccurate or sometimes not available to the defender.

2. The solution’s inflexibility to the change in the estimated attacker’s incentive $I_A$ when repeating the game for a period of time.

Overcoming these shortcomings in the single-stage game is an arduous process [104]. In this chapter, the shortcomings mentioned above are addressed through learning the optimal defense strategy in a repeated game. Online learning is enabled by exploiting the frequent interactions between the jamming attacker and the defending CRN over the frequency spectrum in a repeated game framework.

5.2 System Model

Primarily, in the game-theoretic framework of security problem $\mathcal{G}$, discussed in Chapter 4, each player chooses the strategy that maximizes her payoff assuming the other player is playing optimally. In such settings, the attacker’s preferences are required to be known to the defender to calculate the points of Stackelberg equilibria in $\mathcal{G}$ successfully. The calculated attacker’s response might be inexact (and sometimes concealed) to the defender [76]. Also, assuming a fixed attacker’s incentive $I_A$ during the game which progresses over multiple interactions (game rounds) is not entirely fair [69].

One possible way to address the challenges mentioned above is through adopting the online learning algorithms from the area of machine-learning [105]. The field of online learning focuses on choosing the optimal action, among many, which maximizes the quality of the results (for the learner) in an experiment over many trials. In the context of the security of CRNs, the online learning scheme employs the repeating nature of the attacker/defender interactions over the frequency spectrum. Also, the repeated game structure helps in actualizing the promised reward/punishment as a motivation for the well-behaved/misbehave cognitive users in the CRN. Most importantly, the repeated game structure meets the promise
of penalizing the jamming attacker who was detected due to falling into deployed defense actions (honeypots).

On the other hand, despite being a suitable approach to address the uncertainty about the attacker’s behavior, pure online learning algorithms suffer from the following:

1. A weak initial performance.

2. The inflexibility in considering priori information which might be available from the security experts or other imperfect solutions which might be introduced by game-theoretic based security algorithms.

To this end, six hybrid security algorithms are proposed in this chapter which merge the game-theoretic solutions (which was calculated in Chapter 4) and learning algorithms from the area of machine learning [105]. In other words, the proposed hybrid algorithms use the game-theoretic solution in enhancing the quality of the results which are provided by the online learning algorithms. In particular, the learning algorithms are i) the HEDGE algorithm (in the case of a defender with full feedback information) and ii) the Exponential-weight algorithm for Exploration and Exploitation EXP3 (in the case of a defender with incomplete feedback information).

5.2.1 The Repeated Game Model

The repeated security problem $G_R$ describes the repeated interaction between the CRN defender $D$ and the jamming attacker $A$. Each time step $t$ (i.e., game round), in $G_R$ induces a Stackelberg security game problem $G$, where the CRN defender $D$ commits to a randomized deception strategy $\Sigma_D$ and the jamming attacker $A$, in turn, observes $\Sigma_D$ and chooses the attack strategy $m_j$ which maximizes the expected return for the attacker. In particular, the defender’s goal in $G_R$ is the maximization of her aggregate payoff on the long run. For convenience, the time step $t$ is normalized to Unity in this thesis.
5.2.2 Attacker Behavior Model

As presented earlier in Chapter 4, the attacker(s) is (are) assumed to play optimally as a follower to the defender (the leader) according to the Stackelberg model. The Stackelberg attacker observes the frequency of the deployed defender’s strategy before choosing her optimum attack strategy. This assumption on the attacker’s surveillance capabilities represents the worst case scenario of a very adversarial opponent in security games. The attacker is assumed to be perfectly rational and always respond with the best attack strategy to the defenders deception strategy. The perfect rationality assumption is reasonable as the network attackers are software agents [106–108]. Besides the Stackelberg attacker, celebrate attacker’s models include the Nash attacker, where A plays the expected Nash equilibrium (NE) in the game.

In this chapter, the jamming attacker is assumed to interact with the defending CRN with a high frequency. The frequent interaction assumption is justified by the following:

1. The jamming attacker(s) can detect and track the operating frequency channels of the victim CRN to maximize the expected damage [15].

2. The high repetition rate of the spectrum sensing times (being the arena of the engagement between game players) in CRNs over the target frequency channel/band [16].

Moreover, the attacker’s incentive $I_A$ is assumed to change during repeated game runtime. The change in $I_A$ takes place in the case of a selfish attacker whose concern is on a particular frequency channel and has no interest in other frequency channels. One more possible reason for the change in $I_A$ on the long run is the existence of multiple attackers (cooperated or just scattered over the spectrum) with several attack incentives. This assumption is necessary in forming the worst case scenarios of an adversarial behavior. Also, it clarifies the importance of considering learning during repeated security games in comparison to replicating a fixed game-theoretic solution over time.
5.2.3 Defender Behavior Model

The defender \( D \) owns a set of deception strategies \( h_i \) and plays a mixed strategy profile \( \Sigma_D \) which is a probability distribution over the set of pure deception strategies \( \mathcal{H} \). The defender can calculate the equilibria \( \Sigma_D^* \) in the game through considering approximate attacker behavioral model using the game-theoretic based scheme proposed in Chapter 4. The defender uses hybrid learning algorithms to respond to unknown attacker behaviors through observing the attacker’s best responses in repeated interactions when continuous games are in place.

At the beginning of each game round \( t \), the attacker chooses the best response considering the history of the deployed defender’s deception strategies. The proposed algorithms recommend a particular mixed strategy \( \Sigma_D \) to the defender at each game round. The defender targets the maximization of her cumulated payoff as the game progresses.

Importantly, the defender is not assumed to know about the attacker’s payoffs before playing. However, the game-theoretic solution (game equilibrium) is supposed to be based upon a noisy version of the attacker’s real payoffs.

\[
| \Omega'_A(i, j) - \hat{\Omega}'_A(i, j) | < \epsilon 
\]  

where \( \epsilon \) is the error upper bound, \( 0 \leq \epsilon < 1 \), known to the defender. For simplicity we refer to the error upper bound \( \epsilon \) as the error, henceforth and \( \Omega'_A(i, j) \in [0, 1] \) \( \forall \{i, j\} \) is the rescaled attacker’s payoff, such that

\[
\Omega'_A(i, j) = \frac{\Omega_A(i, j) - \min(\Omega_A(i, j))}{\max(\Omega_A(i, j))} 
\]  

where \( \Omega_A(i, j) \) is calculated earlier in (4.3a).

The feedback structure in the defender’s problem due to the adversarial activities can be of two categories. The first type is the full feedback information, where the defender receives a complete information from all the deception strategies about the whole payoffs after each game round. Put differently; the defender can calculate the regret associated with each defense strategy after each game round. The second type is the partial feedback,
where no additional information beyond the received payoff from the deployed deception strategies is revealed after each game round. These feedback settings formulate a multi-armed bandit (MAB) problem [109]. The MAB problem captures a fundamental predicament whose essence is the trade-off between exploration and exploitation. Sticking with any deception strategy profile $\Sigma_D$ may prevent discovering better strategy profiles, whereas continually seeking a better $\Sigma_D$ will prevent achieving the best total reward from what is known so far. So, to address the feedback types mentioned earlier, the repeated game problem is solved for both categories of the defender’s feedback structure.

To sum up, the assumed available information for the defender is the following:

1. The defender’s action space (deception strategies).

2. The defender’s payoff function $\Omega_D(i,j)$.

3. The defender’s incentive $I_D$ and relative cost factors \{q_1, q_2 and q_3\}.

4. The approximate equilibrium $\Sigma'_D$ from the game theoretic solution calculated in Chapter 4.

Moreover, the defender’s rewards (payoffs) in the proposed repeated security problem $G_R$ are chosen adversarially. Meaning that the attacker chooses the value of the defender’s payoff\(^{18}\).

In this chapter, the defender’s reward is bounded so that, without loss of generality, $r_{(i,j)} \in [0, 1] \ \forall \{i, j\}$, where $r_{(i,j)}$ can be mathematically expressed as:

$$r_{(i,j)} = \frac{\Omega_D(i,j) - \min(\Omega_D(i,j))}{\max(\Omega_D(i,j))} \quad (5.3)$$

where $\Omega_D(i,j)$ is the defender’s payoff as expressed earlier in (4.3b). We simply refer to $r_{(i,j)}$ as $r_i$ henceforth because it is not important to know which attack strategy $m_j$ induced the received reward for the defender.

\(^{18}\)Other types of payoff structure in the literature of online learning include i) the stochastic structure, where the return from a learner’s (defender’s) action is a random variable with a stationary unknown distribution. ii) The fixed structure, where the defender’s rewards are fixed values.
5.3 Online learning in the Deception-based Repeated Security Game Problem

5.3.1 Learning Defender’s Deception Strategy

In the repeated security problem $G_R$, each defender’s deception strategy $h_i$ can be viewed as an expert’s opinion (in the lexicon of machine learning) and the defender’s challenge is to sequentially decide which strategy to choose at each game round (time step). Stated another way, solving the defender’s security problem $G_R$ aims at constructing an online policy which maximizes the defender’s payoff (reward) under multiple interactions with the attacker.

At each game round $t \in \{1, ..., T\}$, the learning algorithm denoted by $B$, chooses a deception strategy $h_i^t$ (possibly random with respect to particular distribution) among $|\mathcal{H}|$ available deception strategies, where $T$ is the total number of game rounds. Then, $A$ responds with launching an attack strategy $m_j^t$, where $j \in \{1, ..., |\mathcal{M}|\}$. This interaction results in an instantaneous reward $r_i^{(B,t)}$ for the defender from deploying deception strategy $h_i^t$, where $i \in \{1, \ldots, |\mathcal{H}|\}$. Thus, the instantaneous reward from learning algorithm ($B$) at time $t$ is

$$R^{(B,t)} = \sum_{i=1}^{|\mathcal{H}|} \sigma_i^t r_i^{(B,t)}$$  \hspace{1cm} (5.4)

where $\sigma_i^t$ is the probability of choosing deception strategy $h_i^t$ at time step $t$.

Consequently, for a sequence of deception strategies $\{h_1^1, h_2^1, \ldots, h_T^T\}$, the cumulative reward from learning algorithm ($B$) after $T$ rounds is:

$$R^B = \sum_{t=1}^T R^{(B,t)} = \sum_{t=1}^T \sum_{i=1}^{|\mathcal{H}|} \sigma_i^t r_i^{(B,t)}$$  \hspace{1cm} (5.5)

which is also called the algorithmic reward.

With the target to analyze the behavior of the learning algorithm, the algorithmic reward $R^B$ may be compared to the optimal fixed deception strategy in hindsight [110, 111]. More specifically, given a sequence of attack strategies $\{m_1^1, m_2^2, \ldots, m_j^T\}$ and the associated sequence of defender’s rewards $r_i^{(B,t)}$ for all deception strategies $h_i^t$ and game rounds $t$, where
henceforth \( r^t_i \), for notational simplicity. The rewards history \( R_{\text{Mem}} \) is:

\[
R_{\text{Mem}} = \begin{pmatrix}
    r^1_1 & r^1_2 & \ldots & r^1_{|\mathcal{H}|} \\
    r^2_1 & r^2_2 & \ldots & r^2_{|\mathcal{H}|} \\
    \vdots & \vdots & \ddots & \vdots \\
    r^T_1 & r^T_2 & \ldots & r^T_{|\mathcal{H}|}
\end{pmatrix}
\]  

(5.6)

Define \( R^T_i \) as the cumulative reward of a deception strategy over all the game rounds \( T \) such that:

\[
R^T_i = \sum_{t=1}^{T} r^t_i
\]

(5.7)

In other words, \( R^T_i \) represents the defender’s reward from playing the same deception strategy \( h_i \) from game round 1 till \( T \).

Given a time horizon \( T \), call \textit{best} as the \textit{best deception strategy} that has the highest cumulative return (sum of assigned rewards) up to time \( T \) with respect to the cumulative rewards of the other deception strategies. Mathematically,

\[
\text{best} := \arg\max R^T_i
\]

(5.8)

Then, the worst-case regret (\( \Psi^B \)) associated with algorithm \( \mathcal{B} \) can be mathematically expressed as:

\[
\Psi^B = R^T_{\text{best}} - R^B
\]

(5.9)

where \( R^B \) is calculated by (5.5). Equation (5.9) measures the defender’s cumulative regrets (from utilizing learning algorithm \( \mathcal{B} \)) had it been able to choose a single deception strategy with a prior knowledge of the whole sequence of attack strategies. The online learning algorithm attempts to minimize the \textit{net loss} \( \Psi^B \).

5.3.2 A Defender with full feedback information

In full information settings, the defender receives a feedback information from all of the deception strategies \( h^t_i \in \mathcal{H} \) at the end of each interaction with the attacker. Put in a different
way, the defender can assess the return she might have got had it been able to choose other deception strategies in the past interaction with the attacker given the attacker’s response. The standard learning algorithm in full information setting is the HEDGE algorithm [79]. Remarkably, the basic algorithm has reincarnated in the literature in different guises, as surveyed in [112].

Algorithm 2 illustrates the utilization of the HEDGE algorithm in the repeated security problem $G_R$. At each time step $t$, a weight $w_t^i$ is assigned to each deception strategy $h_t^i$, where $1 \leq i \leq |H|$. In the initialization step, Algorithm 2 assigns $w_t^0 = Zero$ for all strategies $h_t^i \in H$ (line 1). At each game round $t \in \{1, 2, \ldots, T\}$, first, Algorithm 2 calculates the probability distribution (mixed strategy profile) $\Sigma_t^D = \{\sigma_1^t, \sigma_2^t, \ldots, \sigma_{|H|}^t\}$ using the previous weights of the deception strategies which were calculated in the previous game round (line 3):

$$\sigma_i^t = \frac{\exp(\eta w_{t-1}^i)}{\sum_{j=1}^{|H|} \exp(\eta w_{t-1}^j)}$$  \hspace{1cm} (5.10)

where $\eta$ is the learning parameter. Second, a deception strategy $h_t^i$ is drawn randomly according to the probability distribution $\Sigma_t^D$. Then, the sampled deception strategy $h_t^i$ is deployed by the defender (line 4). Third, the rewards $R_t^i$ are observed for $i = 1, 2, \ldots, |H|$ based upon the attacker’s response (line 5). In the last step, the strategies’ weights $w_t^i$ are updated by the simple multiplicative rule (line 6):

$$w_t^i = w_{t-1}^i + r_t^i \quad \text{for} \quad i = 1, 2, \ldots, |H|$$  \hspace{1cm} (5.11)

Notably, the first term ($w_{t-1}^i$) in (5.11) represents the quality of the results of deception strategy $h_i$ in previous game rounds (hindsight) and the second term ($r_t^i$) describes the return from deception strategy $h_i$ in the current game round.

**Theorem 1** [113] The worst-case regret of the HEDGE algorithm\(^{19}\) ($\Psi_{HED}^\dagger$) at time $T$ with

\(^{19}\)The original theorem was rephrased to consider payoff instead of loss for the learner (defender)
Algorithm 2: The HEDGE Algorithm Framework [113]

**Input:** Parameters: $\eta \in [0, 1]$

**Output:** The defender’s mixed strategy profile $\Sigma_D$

1: Initialization: set $w_i^0 = 0$ for $i = 1, \ldots, |\mathcal{H}|$.
2: for each round $t = 1, 2, \ldots, T$ do
3: Update distribution $\Sigma^t_D = \{\sigma^t_1, \sigma^t_2, \ldots, \sigma^t_{|\mathcal{H}|}\}$ such that
   $$\sigma^t_i = \frac{\exp(\eta w_i^{(t-1)})}{\sum_{j=1}^{|H|} \exp(\eta w_j^{(t-1)})}$$
4: Choose a deception strategy $h_i^t$ according to the distribution $\Sigma^t_D$.
5: Observe the reward vector $\mathcal{R}^t = <r_i^t>$ $\forall i \in \{1, 2, \ldots, |\mathcal{H}|\}$.
6: Set $w_i^t = w_i^{(t-1)} + r_i^t$ for $i = 1, 2, \ldots, |\mathcal{H}|$.
7: end for

**Proof Available in [113].**

Theorem 1 states that the performance of the HEDGE algorithm is almost as good as the best strategy $h_i$ in hindsight. Moreover, the per-round (average) regret $\overline{\Psi}_\text{HED} = \frac{\Psi_\text{HED}}{T} = \frac{\sqrt{2T \ln |\mathcal{H}| + \ln |\mathcal{H}|}}{T} \rightarrow Zero$ as $T \rightarrow \infty$. This means that Algorithm 2 guarantees no per-round regret as the game runs indefinitely for a large number of rounds.

5.3.3 A Defender with limited feedback information

In case of limited (partial) feedback (a.k.a. multi-armed bandit (MAB) problem\(^{20}\)), no additional information beyond the received payoff from the deployed deception strategy is

\(^{20}\)The genesis of the MAB problem lies in the casinos: a gambler must choose one of $k$ non-identical slot machines to play in a sequence of trials. Each machine can yield rewards whose distribution is unknown to the gambler and the gambler’s goal is to maximize his total reward over the sequence. This classic problem captures a fundamental predicament whose essence is the trade-off between exploration and exploitation: Sticking with any single machine may prevent discovering a better machine, whereas continually seeking a better machine will prevent achieving the best total reward from what is known so far.
revealed to the defender after the end of each game round. A standard online learning algorithm in partial (bandit) feedback settings is the Exponential-weight algorithm for Exploration and Exploitation (EXP3) [80]. Due to its weak assumptions about the defender’s feedback structure, the EXP3 algorithm is considered the most pessimistic online learning algorithm [81]. Essentially, the EXP3 algorithm is based on the Hedge algorithm, with the addition of a substantial sampling stride to make an unbiased estimate of the available deception strategies despite being short on having a full information feedback. In particular, Algorithm 3 in this thesis presents the EXP3 algorithm.

**Algorithm 3: The EXP3 Algorithm Framework [80]**

| Input: | Parameters: $\eta \in (0, 1]$ |
| Output: | The defender’s mixed strategy profile $\Sigma_D$ |
| 1: | Initialization: set $w_i^1 = 1$ for $i = 1, \ldots, |H|$. |
| 2: | for each round $t = 1, 2, \ldots T$ do |
| 3: | Update distribution $\Sigma^t_D = \{\sigma_1^t, \sigma_2^t, \ldots, \sigma_{|H|}^t\}$ such that |
| | $\sigma_i^t = (1 - \eta) \frac{w_i^t}{\sum_{j=1}^{|[H]} w_j^t } + \frac{\eta}{|H|} \forall i$ |
| 4: | Choose a deception strategy $h_i^t$ according to the distribution $\Sigma^t_D$. |
| 5: | Observe the reward $r_i^t$ from the $i^{th}$ (deployed) deception strategy. |
| 6: | Employ an estimator $\hat{r}_i^t = <\hat{r}_i^t>$ for the reward vector $\mathcal{R}^t = <r_i^t> \forall i \in \{1, 2, \ldots, |H|\}$ such that: |
| | $r_i^t = \begin{cases} \frac{r_i^t}{\sigma_i(t)}, & \text{if } i = i^t \\ 0, & \text{otherwise} \end{cases}$ |
| 7: | Update weights for all $i = 1, 2, \ldots, |H|$: |
| | $w_i^{(t+1)} = w_i^t \cdot \exp\left(\frac{\eta}{|H|} \hat{r}_i^t\right)$ |
| 8: | end for |

The EXP3 algorithm is a variant of the Hedge algorithm which was described earlier in Algorithm 2. At each time step $t$ of Algorithm 3, a weight $w_i^t$ is assigned to each deception strategy $h_i^t$, where $1 \leq i \leq |H|$. In the initialization step, Algorithm 3 assigns $w_i^1 = Unity$
for all strategies $i \in |\mathcal{H}|$ (line 1). At each game round $t = 1, 2, \ldots$ until game ends at $T$, first, Algorithm 3 calculates the probability distribution (mixed strategy profile) $\Sigma^t_D = \{\sigma^t_1, \sigma^t_2, \ldots, \sigma^t_{|\mathcal{H}|}\}$ as follows:

$$\sigma^t_i = (1 - \eta) \frac{w^t_i}{\sum_{j=1}^{|\mathcal{H}|} w^t_j} + \frac{\eta}{|\mathcal{H}|}$$

where $\eta$ is the learning parameter and at higher values of $\eta$ the EXP3 algorithm plays nearly random. The first term in (5.13) represents the exploitation part while the second term represents the exploration part of Algorithm 3 (line 3). Second, a deception strategy $h^t_i$ is chosen and deployed according to the calculated $\Sigma^t_D$ (line 4). Third, a reward $R^t(i_t, j_t)$ is received based upon the sampled deception strategy $h^t_i$ and the attacker’s strategy $m^t_j$ (line 5). Fourth, An estimator $\hat{R}^t = \frac{R^t(i_t, j_t)}{\sigma^t_i(t)}$ is employed for the deployed deception strategy $h^t_i$. And $\hat{R}^t = 0$ for all other un-deployed deception strategies (line 6). Dividing the observed reward $R$ by $\sigma_i(t)$, the probability that the deception strategy $h^t_i$ was chosen is performed to increase the weight of the deception strategies that were rarely chosen (have smaller weights). The estimated reward vector $\hat{R}^t$ assures that the expectation $\mathbb{E}[\hat{R}]$ equals to the actual reward vector $R$. Finally, the weights for the next game round $w^{(t+1)}_i$ is updated based upon the calculated $\hat{R}^t$ (line 7).

**Theorem 2** [80] *The worst-case regret of the EXP3 algorithm ($\Psi^{HED}$) at time $T$ with parameter $\eta = \sqrt{\frac{|\mathcal{H}| \ln(|\mathcal{H}|)}{(e-1)T}}$ satisfies a higher bound of

$$\Psi^{EXP3} \leq (2\sqrt{e - 1})\sqrt{T|\mathcal{H}|\ln|\mathcal{H}|}$$

(5.14)

Proof Available in [80].

Similarly, the per-round (average) regret $\bar{\Psi}^{EXP3} = \frac{\Psi^{EXP3}}{T} = \frac{(2\sqrt{e - 1})\sqrt{T|\mathcal{H}|\ln|\mathcal{H}|}}{T} \rightarrow Zero$ as $T \rightarrow \infty$. 

89
5.4 The Proposed Hybrid Algorithms

In this section, the combination between the online learning algorithms, presented in Section 5.3, and the game-theoretic solutions, introduced in Chapter 4 is proposed. In particular, three combined algorithms are proposed for the two cases of the defender’s feedback structure; being wholly or partially aware of the behavior of the deception strategies $h_i^t$ at the end of each interaction (game round $t$) with the jamming attacker.

5.4.1 The Hybrid-1 (H1) Algorithm

In the Hybrid-1 (H1) algorithm, the probably inaccurate game-theoretic (GT) solution (calculated in Chapter 4) is used to warm start the online learning algorithm, where the GT solution is fully presented by $\Sigma_D' = \{\sigma'_1, \sigma'_2, \ldots, \sigma'_{|H|}\}$.

To avoid the numerical instability of H1 algorithm in the case of the calculated game equilibrium $\Sigma_D'$ includes a Zero value for some deception strategies $h_i$, we add a minimal value ($10^{-2}$) to the deception strategies’ initial weight $w_{i}^0$. Thus, the initial weights of the H1 algorithm is mathematically expressed as:

$$w_i^0 = \frac{\sigma'_i + (1 \times 10^{-2})}{\sum_{j=1}^{|H|} w_j^0}, \quad i \in \{1, 2, \ldots, |H|\}$$

Note that the H1 algorithm enjoys the same regret upper bound of the underlying online learning algorithm because the initial weights of the deception strategies are still summing to Unity. Algorithm 4 [Algorithm 5] utilizes the Hedge [EXP3] algorithm as a basis in representing a full [bandit] feedback structure.

The line-by-line description of Algorithms 4 and 5 follows the description of the Hedge and EXP3 algorithms described in Section 5.3, respectively, except for the initialization step where $w_i^0$ is set according to (5.15).
**Algorithm 4:** The Hybrid-1 Algorithm in Full Information Feedback ($H1F$)

**Input:** Parameters: set $\eta = \sqrt{\frac{2\ln|\mathcal{H}|}{T}}$

**Output:** The defender’s mixed strategy profile $\Sigma^t_D$ which satisfies a per-round regret of

$$\Psi_{H1F}^t \leq \sqrt{\frac{2T\ln|\mathcal{H}| + \ln|\mathcal{H}|}{T}}$$

1: Initialization: set

$$w^0_i = \frac{\sigma^t_i + (1 \times 10^{-2})}{\sum_{j=1}^{|\mathcal{H}|} w^0_j}, \quad i = 1, \ldots, |\mathcal{H}|$$

(5.17)

2: for each round $t = 1, 2, \ldots T$ do

3: Update distribution $\Sigma^t_D = \{\sigma^t_1, \sigma^t_2, \ldots, \sigma^t_{|\mathcal{H}|}\}$ such that

$$\sigma^t_i = \frac{\exp(\eta w^{(t)}_i)}{\sum_{j=1}^{|\mathcal{H}|} \exp(\eta w^{(t)}_j)}$$

(5.18)

4: Choose a deception strategy $h^t_i$ according to the distribution $\Sigma^t_D$.

5: Observe the reward vector $R^t = \langle r^t_i \rangle \quad \forall i \in \{1, 2, \ldots, |\mathcal{H}|\}$.

6: Set $w^{(t+1)}_i = w^{(t)}_i + r^t_i$ for $i = 1, 2, \ldots, |\mathcal{H}|$.

7: end for
Algorithm 5: The Hybrid-1 Algorithm in Bandit Feedback (H1B)

Input: Parameters: set $\eta = \sqrt{\frac{|\mathcal{H}| \ln(|\mathcal{H}|)}{(e-1)T}}$
Output: The defender’s mixed strategy profile $\Sigma_D^t$ which satisfies a per-round regret of

$$\overline{\Psi}_{H1B} \leq \frac{(2\sqrt{e-1})\sqrt{T|\mathcal{H}| \ln |\mathcal{H}|}}{T}$$  \hfill (5.19)

1: Initialization: set

$$w_0^i = \frac{\sigma_i' + (1 \times 10^{-2})}{\sum_{j=1}^{|\mathcal{H}|} w_0^j}, \quad i = 1, \ldots, |\mathcal{H}|$$  \hfill (5.20)

2: for each round $t = 1, 2, \ldots T$ do
3: Update distribution $\Sigma_D^t = \{\sigma_1^t, \sigma_2^t, \ldots, \sigma_{|\mathcal{H}|}^t\}$ such that

$$\sigma_i^t = (1 - \eta) \frac{w_i^t}{\sum_{j=1}^{|\mathcal{H}|} w_j^t} + \frac{\eta}{|\mathcal{H}|} \quad \forall i$$  \hfill (5.21)

4: Choose a deception strategy $h_i^t$ according to the distribution $\Sigma_D^t$.
5: Observe the reward $r_i^t$ from the $i^{th}$ (deployed) deception strategy.
6: Employ an estimator $\hat{R}^t = < \hat{r}_i^t >$ for the reward vector $\mathcal{R}^t = < r_i^t >$ $\forall i \in \{1, 2, \ldots, |\mathcal{H}|\}$ such that:

$$\hat{r}_i^t = \begin{cases} \frac{r_i^t}{\sigma_i(t)}, & \text{if } i = i^t \\ 0, & \text{otherwise} \end{cases}$$  \hfill (5.22)

7: Update weights for all $i = 1, 2, \ldots, |\mathcal{H}|$:

$$w_i^{(t+1)} = w_i^t \exp\left(\frac{\eta}{|\mathcal{H}|} \hat{r}_i^t\right)$$  \hfill (5.23)

8: end for
5.4.2 The Hybrid-2 (H2) Algorithm

In Hybrid-2 H2 algorithm, GT solution \( \Sigma'_D = \{ \sigma'_1, \sigma'_2, \ldots, \sigma'_{|H|} \} \) is used as an expert advice (a separate suggested deception strategy) for the online learning algorithm. Put differently, we allow the learning algorithm to discover the usefulness of the game-theoretic solution through considering \( \Sigma'_D \) as a new deception strategy. Thus, the initial weights of H2 algorithm are uniformly set as:

\[
 w^0_i = \frac{1}{\sum_{j=1}^{(|H|+1)} w^0_j}, \quad i \in \{1, 2, \ldots, (|H| + 1)\} \quad (5.24)
\]

The H2 algorithm attains a slightly higher regret upper bound in comparison to H1 algorithm because the number of deception strategies in H2 algorithm is increased by one to consider the GT solution. Algorithm 6 [Algorithm 7] utilizes the Hedge [EXP3] algorithm as a basis in formulating the full [bandit] feedback case of H2 algorithm.

5.4.3 The Hybrid-3 (H3) Algorithm

In the Hybrid-3 H3 algorithm, GT solution \( \Sigma'_D = \{ \sigma'_1, \sigma'_2, \ldots, \sigma'_{|H|} \} \) is used as an expert advice and to initialize the online learning algorithm as well.

Algorithm H3 attains a slightly higher regret upper bound in comparison to H1 algorithm because of the added (suggested) deception strategy. Algorithm 8 [Algorithm 9] utilizes the Hedge [EXP3] algorithm as a basis in formulating the full [bandit] feedback case of the H3 algorithm.
Algorithm 6: The Hybrid-2 Algorithm in Full Information Feedback ($H2F'$)

**Input:** Parameters: set $\eta = \sqrt{\frac{2\ln|\mathcal{H}|}{T}}$

**Output:** The defender’s mixed strategy profile $\Sigma_D^t$ which satisfies a per-round regret of

$$\Psi^{H2F} \leq \frac{\sqrt{2T\ln(|\mathcal{H}|+1)} + \ln(|\mathcal{H}|+1)}{T}$$ (5.25)

1: Initialization: set

$$w_i^0 = \frac{1}{\sum_{j=1}^{(|\mathcal{H}|+1)} w_j^0}, \quad i = 1, 2, \ldots, |\mathcal{H}| + 1$$ (5.26)

2: for each round $t = 1, 2, \ldots T$ do

3: Update distribution $\Sigma_D^t = \{\sigma_1^t, \sigma_2^t, \ldots, \sigma_{|\mathcal{H}|}^t, \sigma_{(|\mathcal{H}|+1)}^t\}$ such that

$$\sigma_i^t = \frac{\exp(\eta w_i^t)}{\sum_{j=1}^{(|\mathcal{H}|+1)} \exp(\eta w_j^t)}$$ (5.27)

4: Choose a deception strategy $h_i^t$ according to the distribution $\Sigma_D^t$.

5: Observe the reward vector $\mathcal{R}^t = \langle r_i^t \rangle \quad \forall i \in \{1, 2, \ldots, (|\mathcal{H}|+1)\}$.

6: Set $w_i^{t+1} = w_i^t + r_i^t$ for $i = 1, 2, \ldots, (|\mathcal{H}|+1)$.

7: end for
Algorithm 7: The Hybrid-2 Algorithm in Bandit Feedback ($H2B$)

**Input:** Parameters: set $\eta = \sqrt{\frac{(|\mathcal{H}|+1)\ln(|\mathcal{H}|+1)}{(e-1)^T}}$

**Output:** The defender’s mixed strategy profile $\Sigma_D$ which satisfies a per-round regret of

$$\Psi^{H2B} \leq \frac{(2\sqrt{e-1})\sqrt{T(|\mathcal{H}|+1)\ln(|\mathcal{H}|+1)}}{T}$$

(5.28)

1: Initialization: set

$$w_i^0 = \frac{1}{\sum_{j=1}^{(|\mathcal{H}|+1)} w_j^0}, \quad i = 1, 2, \ldots, |\mathcal{H}| + 1$$

(5.29)

2: for each round $t = 1, 2, \ldots T$ do

3: Update distribution $\Sigma_D^t = \{\sigma_1^t, \sigma_2^t, \ldots, \sigma_{|\mathcal{H}|}^t, \sigma_{(|\mathcal{H}|+1)}^t\}$ such that

$$\sigma_i^t = (1 - \eta) \frac{w_i^t}{\sum_{j=1}^{(|\mathcal{H}|+1)} w_j^t} + \frac{\eta}{|\mathcal{H}| + 1} \quad \forall i$$

(5.30)

4: Choose a deception strategy $h_i^t$ according to the distribution $\Sigma_D^t$.

5: Observe the reward $r_{i^t}^t$ from the $i^t$th (deployed) deception strategy.

6: Employ an estimator $\tilde{R}_i^t = <\tilde{r}_i^t>$ for the reward vector $\mathcal{R}_i^t = <r_{i^t}^t> \quad \forall i \in \{1, 2, \ldots, |\mathcal{H}| + 1\}$ such that:

$$\tilde{r}_i^t = \begin{cases} \frac{r_i^t}{\sigma_i^t}, & \text{if } i = i^t \\ 0, & \text{otherwise} \end{cases}$$

(5.31)

7: Update weights for all $i = 1, 2, \ldots, |\mathcal{H}| + 1$:

$$w_i^{(t+1)} = w_i^t \cdot \exp\left(\frac{\eta \tilde{r}_i^t}{|\mathcal{H}| + 1}\right)$$

(5.32)

8: end for
Algorithm 8: The Hybrid-3 Algorithm in Full Information Feedback (H3F)

**Input:** Parameters: set \( \eta = \sqrt{\frac{2 \ln(|H|+1)}{T}} \)

**Output:** The defender’s mixed strategy profile \( \Sigma_D^t \) which satisfies a per-round regret of

\[
\bar{\Psi}^{H3F} \leq \frac{\sqrt{2T \ln(|H|+1)} + \ln(|H|+1)}{T} \tag{5.33}
\]

1: Initialization: set

\[
w_i^0 = \frac{\sigma'_i + (1 \times 10^{-2})}{\sum_{j=1}^{|H|+1} w_j^0}, \text{ for } i = 1, 2, \ldots, (|H|+1) \tag{5.34}
\]

2: **for** each round \( t = 1, 2, \ldots, T \) **do**

3: Update distribution \( \Sigma_D^t = \{\sigma^t_1, \sigma^t_2, \ldots, \sigma^t_{|H|}, \sigma^t_{(|H|+1)}\} \) such that

\[
\sigma^t_i = \frac{\exp(\eta w_i^{(t)})}{\sum_{j=1}^{|H|+1} \exp(\eta w_j^{(t)})} \tag{5.35}
\]

4: Choose a deception strategy \( h_i^t \) according to the distribution \( \Sigma_D^t \).

5: Observe the reward vector \( R^t = \langle r_i^t \rangle \quad \forall i \in \{1, 2, \ldots, (|H|+1)\} \).

6: Set \( w_i^{(t+1)} = w_i^{(t)} + r_i^t \) for \( i = 1, 2, \ldots, (|H|+1) \).

7: **end for**
Algorithm 9: The Hybrid-3 Algorithm in Bandit Feedback ($H3B$)

**Input:** Parameters: set $\eta = \sqrt{\frac{(|\mathcal{H}|+1) \ln(|\mathcal{H}|+1)}{(e-1)T}}$.

**Output:** The defender’s mixed strategy profile $\Sigma^t_D$ which satisfies a per-round regret of

$$\Psi^{H3B} \leq \frac{(2\sqrt{e-1})\sqrt{T(|\mathcal{H}|+1) \ln(|\mathcal{H}|+1)}}{T}$$

(5.36)

1: Initialization: set

$$w^0_i = \frac{\sigma^0_i + (1 \times 10^{-2})}{\sum_{j=1}^{|\mathcal{H}|+1} w^0_j}, \text{ for } i = 1, 2, \ldots, (|\mathcal{H}|+1)$$

(5.37)

2: for each round $t = 1, 2, \ldots T$ do

3: Update distribution $\Sigma^t_D = \{\sigma^t_1, \sigma^t_2, \ldots, \sigma^t_{|\mathcal{H}|}, \sigma^t_{(|\mathcal{H}|+1)}\}$ such that

$$\sigma^t_i = (1 - \eta) \frac{w^t_i}{\sum_{j=1}^{|\mathcal{H}|+1} w^t_j} + \frac{\eta}{|\mathcal{H}|+1} \quad \forall i$$

(5.38)

4: Choose a deception strategy $h^t_i$ according to the distribution $\Sigma^t_D$.

5: Observe the reward $r^t_i$ from the $i^{th}$ (deployed) deception strategy.

6: Employ an estimator $\hat{\mathcal{R}}^t = <\hat{r}^t_i>$ for the reward vector $\mathcal{R}^t = <r^t_i> \forall i \in \{1, 2, \ldots, |\mathcal{H}|+1\}$ such that:

$$\hat{r}^t_i = \begin{cases} \frac{r^t_i}{\sigma^t_i}, & \text{if } i = i^t \\ 0, & \text{otherwise} \end{cases}$$

(5.39)

7: Update weights for all $i = 1, 2, \ldots, |\mathcal{H}|+1$:

$$w^{(t+1)}_i = w^t_i \cdot \exp\left(\frac{\eta \hat{r}^t_i}{|\mathcal{H}|+1}\right)$$

(5.40)

8: end for
5.5 Simulation Results

In this section, the results of a Matlab-based simulation of the repeated security game problem $G_R$ between the jamming attacker and the defending CRN is presented.

The simulation process is performed to compare the performance of the proposed hybrid algorithms H1, H2 and H3 to the behavior of the two standard online learning algorithms in the literature of machine learning. In particular, i) the HEDGE algorithm (in the case of a defender with full feedback information) and ii) the Exponential-weight algorithm for Exploration and Exploitation EXP3 (in the case of a defender with partial feedback information).

The simulation is run against two celebrate types of attackers i) an attacker who samples her actions from a fixed probability distribution which represents the mixed strategy NE in the single-run game $G$ which was analyzed in Chapter 4. ii) A Stackelberg attacker who adopts her actions according to the frequency of the defender’s choices in the past (the worst-case attacker). Finally, the simulation examines the case when the attacker’s incentive changes during the $G_R$ time.

The simulation setup is as follows:

1. The number of game rounds $T$ is set to 60, this represents approximately 6 seconds of repeated play\(^2\) if the game players engage on every consequent CRN’s sensing cycle.

2. Each experiment is repeated 10,000 times to compensate for the mixed deception/attack strategies utilized by the game players and to get more reliable results.

\(^2\)The IEEE 802.22 CRN schedules the QP every 100ms on the average. Thus ten interactions a second are expected with the attacker. In addition, the IEEE 802.22 standard requires the CRN to evacuate the channel within few seconds upon detecting the PU’s signal. Thus, in the worst case, the decision about the attacker’s activity over the sensed channel should be made within few seconds too.
Table 5.1: Typical values for the learning rate $\eta$ of the proposed hybrid algorithms and the standard learning algorithms when $T = 60$.

<table>
<thead>
<tr>
<th></th>
<th>HEDGE</th>
<th>EXP3</th>
<th>H1F</th>
<th>H1B</th>
<th>H2F</th>
<th>H2B</th>
<th>H3F</th>
<th>H3B</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\eta$</td>
<td>0.2316</td>
<td>0.2794</td>
<td>0.2316</td>
<td>0.2794</td>
<td>0.2444</td>
<td>0.3229</td>
<td>0.2444</td>
<td>0.3229</td>
</tr>
</tbody>
</table>

3. Without loss of generality, $I_D$, $I_A$, and $\phi$ are set to 40, 40 and 10, respectively. These relatively high values of the players’ defense and attack incentives are arbitrarily chosen to motivate the players to engage in the game.

4. Unless otherwise specified, the learning parameter $\eta$ is hand tuned as described in the initialization step of each of the proposed algorithms $H1F$, $H1B$, $H2F$, $H2B$, $H3F$ and $H3B$ and the standard learning algorithms HEDGE and EXP3 described earlier. Table 5.1 illustrates the typical values of $\eta$ for each of the utilized algorithms in this section.

The results presented in the following include the average per-round regret $\Psi$ and the cumulative regret of the standard version of the adopted learning algorithms (i.e., the HEDGE and the EXP3 algorithms) which are compared to the behavior of the average per-round regret and the cumulative regret of the proposed hybrid algorithms. The GT solution is utilized in the proposed algorithms as follows:

1. The H1 algorithm initializes the learning algorithm with the GT solution, labeled in the graphs by $H1F$ and $H1B$ for full and bandit feedback, respectively.

2. The H2 algorithm suggests a separate deception strategy based upon the GT solution, labeled in the graphs by $H2F$ and $H2B$ for full and bandit feedback, respectively and

3. The H3 algorithm combines both H1 and H2 algorithms (initialization and suggestion), labeled in the graphs by $H3F$ and $H3B$ for full and bandit feedback,
respectively.

5.5.1 A Defender with Full Feedback Information

Figure 5.1 and Figure 5.2 compare the performance of the standard version of the HEDGE algorithm and the proposed hybrid algorithms ($H1F$, $H2F$ and $H3F$) against an attacker who plays a mixed strategy Nash equilibrium and a pure strategy Stackelberg equilibrium, respectively. In Figures 5.1 and 5.2, the x-axis shows the game round $t$ and the y-axis depicts the per-round regret $\Psi$ in the case of a defender with full information feedback. In both Figures 5.1 and 5.2, the graphs labeled (a), (b), (c), and (d) show the evolution of the learning process when the error $\epsilon$ equals 0, 0.1, 0.2 and 0.3, respectively. As clear from Figures 5.1 and 5.2, the average regret of the proposed algorithms decreases (in general) as the game progresses, irrespective of the error value in the integrated GT solution. The reason behind the reduced regret is the intrinsic characteristic of the learning algorithms where at each time step higher weights are assigned to the deception strategies that performed better in the past. Thus, if the attacker is following any rational sequence of attack strategies (i.e., there is something to learn about the attacker’s choices), the defender can learn the optimum deception strategies as the game progresses.

Interestingly, the value of $T$ which represents the game horizon is an application-specific parameter\footnote{As discussed earlier, $T$ is chosen based upon the expected number of interactions between the game players. The game horizon $T$ varies from few second to few years. In some applications, the game horizon $T$ is indefinite, creating a game with no predetermined length [114].} and is used to adjust the learning factor $\eta$ in order to achieve a specific guaranteed regret upper bound at the game end.

In Figure 5.1, when the error is less than or equals to 0.1, algorithms $H1F$, $H2F$ and $H3F$ outperform the standard HEDGE algorithm when playing against a Nash attacker. At higher errors, i.e. $\epsilon > 0.1$, the proposed algorithms show a better initial performance and a competitive tendency to reduce the regret as the game progresses in comparison to the
Figure 5.1: The behavior of the proposed hybrid algorithms in full information feedback vs. Nash attacker for different values of error (\(\epsilon\)).

standard HEDGE algorithm. Importantly, this result is a direct consequence of combining the possibly inaccurate GT solution with the online learning algorithms in the full information feedback structure.

Notice that the initial poor performance of the \(H2F\) algorithm in comparison to \(H1F\) and \(H3F\) algorithms is because of the uniform random start of \(H2F\) algorithm. Beyond the first 20 rounds, algorithms \(H2F\) and \(H3F\) hold a relatively higher regret in comparison to \(H1F\) and HEDGE algorithms because they have an extra acclaimed deception strategy (the GT solution) which negatively affects the theoretical average regret upper bound as discussed earlier. Moreover, beyond the first 20 rounds, the behavior of the \(H3F\) algorithm
Figure 5.2: The behavior of the proposed hybrid algorithms in full information feedback vs. Stackelberg attacker for different values of error ($\epsilon$)

is relatively the worst among the proposed algorithms because in the $H3F$ algorithm the erroneous GT solution contributes to the initial weights of the deception strategies besides being considered as a separate deception strategy.

In Figure 5.2, the performance of algorithms $H1F$, $H2F$ and $H3F$ is evaluated when the attacker is playing adversarially with Stackelberg attack strategies. The Stackelberg attacker always plays the optimal attack strategy after observing the frequency of each of the defender’s choices, presenting the worst-case attack attempt.

Also in Figure 5.2, the average regrets $\Psi$ of the proposed algorithms and the standard HEDGE algorithm are relatively higher when playing against a Stackelberg attacker compared
to playing against a Nash attacker. The average regret of the proposed algorithms tends to
decrease over time as the defender learns more about the attacker’s behavior. Finally, in
Figure 5.2, the $H1F$ algorithm gives the best results and it is quite stable over game rounds
because of its relatively good start (compared to $H2F$) and lower expected regret upper
bound (in comparison to $H3F$).

The usefulness of the proposed hybrid algorithms ($H1F$, $H2F$ and $H3F$) is pointed out
in Figures 5.1(a) and 5.2(a) where the error in the calculated GT solution is Zero ($\epsilon = 0$).
Apparently, in this case, all of the proposed algorithms outperform the standard HEDGE
algorithm over the game rounds. The HEDGE algorithm always starts with a per-round
regret $\Psi^{\text{Hedge}}$ of approximately 23.5% [26%] in comparison to the best fixed strategy in
hindsight against Nash [Stackelberg] attacker. $H1F$, $H2F$ and $H3F$ achieve 91.4% [88.4%],
20% [11.5%] and 92% [92%] enhancement in comparison to $\Psi^{\text{Hedge}}$ against Nash [Stackelberg]
attacker, respectively. Beyond the first 20 rounds, $H1F$, $H2F$ and $H3F$ perform close to
the HEDGE algorithm or even better when the error $\epsilon \leq 0.3$.

Definitely, the proposed combined algorithms solve the poor initial performance of the
pure HEDGE algorithm. Note that the aforementioned gains are scenario-specific (i.e. depend
on $I_D$, $I_A$ and $\phi$) thus rely on the calculated game equilibrium.

5.5.2 A Defender with Bandit Feedback

Broadly, when playing a repeated security game with bandit feedback settings, the average
regret is expected to be higher in comparison to the case when full feedback settings are in
place. The reason behind this is, in bandit feedback the defender (learner) only receives a
feedback information from the deception strategies which were deployed in the same game
round. Thus, the defender continuously tends to explore other deception strategies for a
better payoff. The average regret is relatively higher due to the exploration property of the
bandit-based learning algorithms.

The results in Figure 5.3 and Figure 5.4 evaluate the performance of the proposed hybrid
algorithms \((H1B, H2B \text{ and } H3B)\) in comparison to the performance of the standard version of the EXP3 algorithm. The attacker in this experiment plays a mixed strategy Nash equilibrium from in Figure 5.3 and a pure strategy Stackelberg equilibrium in Figure 5.4. The x-axis on Figures 5.3 and 5.4 shows the game round \(t\) and the y-axis depicts the per-round regret \(\Psi\) for a defender with bandit feedback. Similarly, in both Figures 5.3 and 5.4, the graphs labeled (a), (b), (c), and (d) show the numerical results when the error \(\epsilon\) equals 0, 0.1, 0.2 and 0.3, respectively.

It is clear from the results in Figures 5.3 and 5.4 that \(H1B, H2B \text{ and } H3B\) algorithms indicate the tendency to reduce the per-round regret as the game progresses even at higher values of \(\epsilon\) due to learning the attacker’s behavior. In particular, algorithms \(H1B, H2B \text{ and } H3B\) outperforms the standard EXP3 algorithm in the first 20 rounds against the Nash attacker with error values up to 30\%. Beyond 20 rounds, the \(H1B\) algorithm outperforms the other learning algorithms due to its excellent initial performance and lower expected regret upper bound. Moreover, when higher values of error \(\epsilon\) are expected, \(H2B \text{ and } H3B\) algorithms perform similar/worse than \(H1B \text{ and } \) EXP3 algorithms beyond the first 20 rounds because of the higher expected regret upper bound of \(H2B \text{ and } H3B\) algorithms.

As explained before, Algorithms \(H1B\) and \(H3B\) have a better initial performance than Algorithms \(H2B\). However, Algorithms \(H1B\) and \(H3B\) have a relatively higher average regret in the following game rounds with respect to \(H1B\) Algorithm. One more observation is that \(H3B\) Algorithm is the most sensitive among all the proposed hybrid algorithms to the increase in the error \(\epsilon\). The reason is in \(H3B\)’s utilization of the erroneous GT solution \(i)\) in initializing the algorithmic weights of the deception strategies and \(ii)\) using a suggested separate deception strategy.
Figure 5.3: The behavior of the proposed hybrid algorithms in Bandit feedback vs. Nash attacker for different values of error ($\epsilon$)
One more observation is the oscillating behavior of the learning processes which are based on the EXP3 algorithm in Figures 5.3 and 5.4 in comparison to the smooth learning curves which are based on the HEDGE algorithm depicted in Figures 5.1 and 5.2. The reason behind the oscillating learning curves in the EXP3-based algorithms is the embedded exploration from the property of the non-received feedback information of the nondeployed deception strategies. In Figure 5.4, the performance of the learning processes of algorithms $H1B$, $H2B$ and $H3B$ are examined when the attacker is playing according to the Stackelberg model. Note that the case in Figure 5.4 is the most difficult scenario for online learning algorithms where a defender with limited information plays against a knowledgeable attacker. Thus, this game scenario is used as a proof on the robustness of the proposed hybrid algorithms.

The average regrets $\Psi$ of $H1B$, $H2B$, $H3B$ and EXP3 algorithms from playing against a Stackelberg attacker are higher than the average regret experienced when playing against a Nash attacker at round 1 because of the very adversarial nature of the knowledgeable Stackelberg attacker. In addition, the EXP3-based learning algorithms hold a higher regret upper bound in comparison to the HEDGE-based algorithms due to the exploration property of the EXP3-based learning algorithms.

Moreover, the exploration property of algorithms $H1B$, $H2B$ and $H3B$ might be confusing for the Stackelberg attacker. Put differently, the defender might choose a deception strategy that returns a higher regret against an adaptive attacker because of the exploration nature of the underlying learning algorithm. This explains the increasing average regret of the $H2B$ and $H3B$ algorithms after the very good start at round 1 in Figure 5.4. Yet, the average regret $\Psi$ of the proposed algorithms is enhanced (in general) as the game progresses because of the increase in defender knowledge about the effect of the deception strategies.

Figures 5.3(a) and 5.4(a) illustrate the effectiveness of the $H1B$, $H2B$ and $H3B$ algorithms as the defender utilizes the GT solution with $\epsilon = 0$. In the error-less case, $H1B$, $H2B$ and $H3B$ algorithms outperform the standard EXP3 algorithm over the entire game.
Figure 5.4: The behavior of the proposed hybrid algorithms in bandit feedback vs. Stackelberg attacker for different values of error ($\epsilon$)
rounds.

To illustrate the usefulness of $H1F$, $H2F$ and $H3F$ (and also $H1B$, $H2B$ and $H3B$) algorithms, Figure 5.5 shows the cumulative regret $\Psi$ of the proposed algorithm \(i\) in the case of a full feedback (top plots), \(ii\) in the case of a bandit feedback (bottom plots), \(iii\) in the case of Nash attacker (left plots) and finally \(iv\) in the case of Stackelberg attacker (right plots).

Two important observations from the results in Figure 5.5. First, the cumulative regret always increases because the learning algorithms are always searching for better solutions even if they reach to a near-optimum solution. Second, the rate of the cumulative regret $\Psi$ is higher when the defender plays the bandit feedback game in comparison to the full feedback game because of the exploration property of the bandit-based learning algorithms.

Finally, Figure 5.6 shows the comparative performance of $H1F$, $H2F$ and $H3F$ (and also $H1B$, $H2B$ and $H3B$) algorithms when the attacker changes her behavior during game time. Specifically, when the attacker’s incentive $I_A$ drops from 40 to 20 at game round number 150 as depicted by the thick dashed vertical line.

Obviously, $H1F$, $H2F$ and $H3F$ (and also $H1B$, $H2B$ and $H3B$) algorithms were able to adapt to the change in the attacker’s behavior. $H1B$ [$H2F$] algorithm is the fastest to converge in the bandit [full] feedback settings after the attacker’s behavioral change at round number 150 as shown in Figure 5.6(b). The behavior $H1F$, $H2F$ and $H3F$ (and also $H1B$, $H2B$ and $H3B$) algorithms depend on the internal weight of the deception strategies within each algorithm and the regret value due to the attacker’s behavioral change. The results in Figure 5.6 displays the resilience of the proposed combined learning algorithms to the change in the attacker’s behavior. Thus, Figure 5.6 proves the usefulness of adapting the learning algorithms in the repeated security games. Otherwise, the defender would have played the same static GT solution over the repeated game rounds, causing either a waste in the defense resources (if extra defense is deployed) or an increased probability of losing the frequency
channel (if less defense is used).
Figure 5.5: The cumulative regret of the proposed hybrid algorithms with $\epsilon = 0$ for: (a) Defender with full feedback vs. Nash attacker, (b) Defender with full feedback vs. Stackelberg attacker, (c) Defender with bandit feedback vs. Nash attacker and (d) Defender with bandit feedback vs. Stackelberg attacker.
Figure 5.6: The behavior of the proposed hybrid algorithms in full information feedback and in partial information feedback when the attacker’s behavior changes.
5.6 Chapter Summary

In this chapter, six hybrid algorithms which combine both the advantages of the game theoretic solutions and the online learning algorithms in a repeated security game framework are introduced. The proposed hybrid algorithms have a theoretical regret upper bound and enjoy an excellent initial behavior with respect to celebrated standard online learning algorithms in both cases of defender’s feedback structure. The proposed hybrid algorithms were tested first against an attacker who plays a mixed strategy Nash equilibrium and then against a knowledgeable attacker who plays a pure Stackelberg equilibrium. And finally, against an attacker whose behavior changes during game time. All of the proposed algorithms outperformed the standard learning algorithms over the game course when the error in the calculated game equilibrium is less than 10%. In addition, the proposed algorithms achieved up to 92% decrease in the initial per-round regret in comparison to the standard learning algorithms.
Chapter 6

Conclusions

In Section 6.1, a summary of the work in this thesis is presented along with the engineering significance and thesis conclusions. In Section 6.2, suggested future work is presented.

6.1 Thesis Summary and Conclusions

The main objective of this thesis is to propose a solution based on the deception tactics to protect the cognitive radio networks from the contingent acute jamming attacks. Generally speaking, both of the theoretical analysis and the numerical results demonstrate the usefulness of the proposed deception-based defense mechanism in reducing the probability of denial of service from the contingent acute jamming attacks even if the attacker’s payoff function is unknown to the defender.

Specifically, to accomplish the above-mentioned thesis objective, first, a security threat assessment for the cognitive radio network was performed under the assumption of multiple denial-of-service (DoS) attacks. The security threat assessment process indicated a 51.3% increase in the severity DoS threats when the attackers collude in comparison to the most severe sole DoS attack. Second, a set of deception based defense strategies is introduced and their effectiveness against the contingent acute jamming attacks is investigated in a game theoretic framework. When a CRN defender utilizes the deception tactics, she, the defender, can reduce the severity of the deceiving attack to nearly 0% irrespective of the PU activity over the targeted frequency channel.

Finally, a set of hybrid online learning algorithms which combines the pure online learning approach with the solution calculated by the game theory is proposed for the defending CRN when dealing with unknown attackers’ models or behavior in a repeated game frame-
work. The simulation results showed up to 92% reduction in the per-round regret when the defender uses the proposed hybrid algorithms in comparison to celebrated pure online learning schemes. In addition, the numerical results illustrated the dependency of the achievable reduction in the per-round regret on the accuracy of the calculated game theoretic solution in different game scenarios.

In Chapter 3, the security threat assessment of the IEEE 802.22 networks is performed under the assumption of a very hostile environment where multiple attackers cooperate to inflict the maximum damage on the victim cognitive network. Unlike the previous works in the literature, Chapter 3 addresses the challenge mentioned above through using the holistic approach of assessing the combined effects of the DoS attacks. The Bayesian Attack Graph (BAG) model is utilized to capture the probabilistic dependencies among the IEEE 802.22 DoS threat–environment and known cognitive radio network’s vulnerabilities.

Chapter 3 introduces the BAG model representation as a single and sufficient security metric for the cognitive radio networks. The BAG model is used to calculate the DoS probability of simultaneous multiple attack scenarios and the probability of exploiting known vulnerabilities of IEEE 802.22. Thus, chapter 3 pinpoints the most likely DoS attack paths (attack strategies) in the IEEE 802.22 networks.

The simulation results indicate up to 51.3% increase in the probability of DoS in the IEEE 802.22 networks considering simultaneous multiple attacks in comparison to the most severe sole attack, proofing the importance of addressing the effect of combined attacks. Moreover, the simulation results proved the importance of protecting the spectrum sensing process being a prime target for the attackers, where manipulating the onboard sensing circuitry and the reception of the spectrum sensing reports/decision was victimized in approximately 40% of the attack scenarios.

By and large, the results presented in Chapter 3 proves the usefulness of the BAG model as a feasible CR vulnerability metric that can facilitate the creation of the security tightening
plan by network engineers. To our best knowledge, this is the first work that introduces the BAG model as a quantitative security metric in cognitive radio networks.

In Chapter 4, the solution to the problem of the contingent acute jamming attacks (the deceiving attack) is introduced in a Stackelberg game theoretic framework. The deceiving attack deceives the victim cognitive radio network by manipulating the CR’s onboard sensing circuitry and the receiving circuitry during the reporting times of the spectrum sensing results and the spectrum decision with a target to falsify the legitimate PU’s activity over the sensed channel/band. To the author’s best knowledge, no previous works in the literature identified/defined the deceiving attack or investigated its impact on cognitive radio networks.

Chapter 4 introduces a set of defense strategies based on the deception tactics as a solution to the deceiving attack through a Stackelberg deception based defense mechanism which could decrease the probability of success of the deceiving attacks to nearly 0% when the defender has a high incentive to protect the channel.

The Stackelberg assumption formulates the worst case adversarial behavior where the attacker chooses the optimal attack action after observing the frequency of the deception actions deployed by the defender in hindsight. The game solution (the game equilibrium, i.e., a particular deployment probability of deception strategies) is calculated such that no player can achieve a higher payoff by unilaterally changing from the calculated game equilibrium.

Chapter 4 presents the derivation of the closed-form expression for the Stackelberg equilibrium when the PU activity pattern is common knowledge in the game. The game theoretic solution calculated in chapter 4 is sensitive to the accuracy of the assumed attacker’s model or behavior.

Chapter 5 addresses the sensitivity of the game equilibrium (which was calculated in chapter 4) to the quality of the utilized attacker’s model through elevating the deception-based defense mechanism by considering online learning in a repeated game framework. In particular, the defender learns to choose the optimum deception strategy by assessing the re-
ceived feedback after each interaction with the deceiving attacker. Towards this end, chapter 5 proposes a set of hybrid online learning algorithms which combines both the advantages of the (possibly inaccurate) game theoretic solution and the online learning schemes in cases when the defender receives a full feedback or a partial (bandit) feedback at the end of each game round.

Specifically, six hybrid algorithms which enjoy a good theoretical regret upper bound and an excellent initial behavior with respect to the celebrated standard online learning algorithms are introduced. The simulation results affirm that the proposed hybrid algorithms outperform the famous standard learning algorithms (the HEDGE algorithm in the case of a defender with full feedback information and the Exponential-weight algorithm for Exploration and Exploitation EXP3 in the case of a defender with partial feedback information) over the game course when the error in the estimated game equilibrium is limited.

In addition, the proposed hybrid algorithms were successfully tested first against the Nash attacker who cannot observe the defender’s actions and second, against the worst case knowledgeable attacker who plays the Stackelberg equilibrium. The simulation results show that the behavior of the proposed hybrid algorithms is better than the standard learning algorithms when the error in the estimated game equilibrium is less than 10%. Also, the proposed hybrid algorithms achieve up to 92% decrease in the initial per-round regret in comparison to the standard learning algorithms in the simulated game scenarios.

Engineering Significance

The proposed research investigates the important and non-trivial problem of assessing and mitigating the coordinated multiple jamming attacks in cognitive radio networks. In this context, the proposed research provided a rigorous understanding of the security vulnerabilities and probable security tightening measures of a significant part of the future wireless networks, CRNs. Thus, the future availability of CRN technology in the market highly
depends on countering any probable misbehaving users.

The deceiving attack is an exclusive and dominant attack to CRNs that can cause a severe denial of service (DoS) to the entire network. Accordingly, the mitigation of the deceiving attack raises a great challenge to CRNs especially when the attacker’s behavioral model is unknown.

Particularly, the design of the deception-based defense mechanism enables the mitigation of sophisticated and dynamic multiple coordinated jamming attacks through the selection of optimum deception strategies. Besides, the proposed security mechanism is resilient to the change in the attackers’ behavior and the errors in the estimated attacker’s model.

6.2 Thesis Limitations and Suggestions for Future Work

The work presented in this thesis has a significant potential for future research. Some thesis limitations and suggestions for the future work include:

1. Deception-based defense mechanism prototyping and testing:
   Setting up a testbed of a practical CRN with different types of devices as primary and secondary users. The planned CRN testbed includes different kinds of CRNs’ security mechanisms, specifically: a) data security mechanisms, which protect the confidentiality, integrity and authenticity of the communication data. b) Primary user security mechanisms, which guarantee the PU’s rights in conducting an interference-free communication and c) cognitive security mechanisms, which target protecting the secondary users’ rights from misbehaving or malicious activities, such as the proposed deception-based defense mechanism.

   Primary challenge in the design of the CRNs’ testbed is the integration of the security mechanisms with the protocol reference model (PRM) of the CRNs, such as the IEEE 802.22 PRM introduced in [2]. The planned CRN testbed
would help the security engineers in the testing and rapid-prototyping of CRNs security mechanisms.

2. Integrating the proposed defense mechanism with an intrusion detection system (IDS):

Combine the proposed work with an intrusion detection system (IDS) to provide a real-time gathering of critical cognitive network parameters such as primary user access time, packet delivery ratio (PDR), received signal strength (RSS), etc. The integration of the IDS aims at detecting new (i.e., not known beforehand) abnormal behavior in the cognitive system, accordingly, adjust the deception actions. The main challenge would be in the investigation/identification of the thresholds of the critical cognitive network parameters below which a network activity would be considered malicious. The expected benefit from such an integration is in producing a resilient deception-based defense mechanism.

3. Another future direction is viable through considering more realistic attack situations, specifically:

a) In the proposed game formulation we assume that the attackers collude against the defending CRN to formulate the worst case scenario for the defender. Though, more investigation on the case when the jamming attackers compete over the targeted frequency channel/band might be useful. The new formulation of the game problem brings up a more precise result in such a scenario. The main challenge lies in the formulation of the security problem under the assumption of possible collisions among attackers over the targeted frequency channel/band and the impact of these collisions on the behaviors of the defender and the attackers. Again, the game theory is the candidate mathematical tool to tackle such a security problem.
b) In the proposed work, the attacker is assumed to be oblivious, in the sense that she does not learn from the previous interactions with the defender. The new research question would be, is it possible to design a defense mechanism which maximizes the defender’s payoff as the game runs against a non-oblivious attacker? It is worth noting that the notion of regret has no meaning when both players (the attacker and the defender) are learning each other’s behavior. Thus, a new metric should be designed and used when dealing with such a non-oblivious attacker.
Bibliography


[77] M.-F. Balcan, A. Blum, N. Haghtalab, and A. D. Procaccia, “Commitment without


[85] C. Zhao, W. Wang, L. Huang, and Y. Yao, “Anti-PUE attack based on the transmitter fingerprint identification in cognitive radio,” in *5th International Conference on Wire-


and Networking, June 2017 (Under Review).


