Development of Spectrum Sensing Cooperation and Fusion Strategies for Tactical Heterogeneous Networks in Mobile Environments

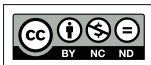
by

Bryan GINGRAS

THESIS PRESENTED TO ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
IN PARTIAL FULFILLMENT OF A MASTER'S DEGREE
WITH THESIS IN ELECTRICAL ENGINEERING
M.A.Sc.

MONTREAL, APRIL 2ND, 2020

ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC



BOARD OF EXAMINERS

THIS THESIS HAS BEEN EVALUATED

BY THE FOLLOWING BOARD OF EXAMINERS

M. Georges Kaddoum, Thesis Supervisor
Department of Electrical Engineering, École de technologie supérieure

M. Eric Granger, President of the Board of Examiners
Department of Systems Engineering, École de technologie supérieure

M. Michel Kadoch, Member of the jury
Department of Electrical Engineering, École de technologie supérieure

THIS THESIS WAS PRESENTED AND DEFENDED

IN THE PRESENCE OF A BOARD OF EXAMINERS AND THE PUBLIC

ON MARCH 12TH, 2020

AT ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

ACKNOWLEDGEMENTS

I would like to begin by thanking my supervisor, Professor Georges Kaddoum, whose support, guidance and counsel were priceless in completing this endeavour.

To Professor François Gagnon, thank you for being my supervisor in the earlier stages of my degree. I've learned so much from you, and am very grateful. I wish you great success with your new responsibilities, you have my support.

To Simon, Ali, Mohammed Ahmad, Mohammed Albi, and Diala, I say thank you for your patience and your generosity of time and spirit. To Marwan, thank you for being a source of encouragement for so many years, both at McGill and at ÉTS.

To my parents, my sister, and my nieces, thank you for always supporting me in my desire to pursue this degree. To my brother Benjamin, thank you for setting an example and a high standard for me to follow.

To everyone at the Fraternité du Piranha, thank you for welcoming me and offering me such a vast network of friends and colleagues.

I would also like to thank the members of my jury for taking the time to evaluate this dissertation.

Finally, I wish to extend my sincerest thanks to everyone behind the Ultra Electronics TCS NSERC Chair whose support was invaluable to this research project.

Développement de stratégies de coopération et de fusion de détection de spectre pour des réseaux tactiques hétérogènes dans des environmements mobiles

Bryan GINGRAS

RÉSUMÉ

La détection du spectre dans les réseaux tactiques sans fil qui sont attaqués par des brouilleurs est une considération importante pour assurer la sécurité et l'efficacité du personnel militaire sur le terrain. Il est nécessaire que les membres de ces réseaux sachent quels canaux du spectre sont compromis et lesquels sont sécuritaires à utiliser pour la transmission de données. Les émetteurs sans fil qui composent ces réseaux peuvent y parvenir en détectant le niveau d'énergie sur différents canaux afin de déterminer s'il y a des brouilleurs actifs sur ces canaux. Ils peuvent ensuite partager ces informations avec leurs pairs afin d'identifier et d'éviter les brouilleurs de façon collaborative. Il existe plusieurs solutions basées sur l'apprentissage par renforcement qui permettent aux émetteurs sans fil d'élaborer une politique de transmission basée sur leurs observations de l'activité des brouilleurs, mais ces solutions échouent souvent lorsque le comportement d'un brouilleur est aléatoire, empêchant ainsi les algorithmes d'apprentissage par renforcement d'apprendre et d'anticiper leurs actions.

Dans cette thèse, nous discutons d'abord de la détection collaborative du spectre et de la théorie derrière les radios cognitives, le brouillage et l'anti-brouillage. Ensuite, nous détaillons le système considéré pour représenter le problème d'anti-brouillage à plusieurs agents. Nous introduisons ensuite un algorithme collaboratif pseudo-aléatoire de sélection de canal et un schéma de collaboration et de fusion de données basé sur des vecteurs de super-décision afin d'améliorer la connaissance par rapport à l'utilisation du spectre à travers le réseau. Les résultats des simulations montrent que cette solution mène à des taux plus élevés de brouilleurs détectés ainsi qu'à une augmentation du nombre de transmissions qui ont lieu sur des canaux non brouillés.

Mots-clés: Détection collaborative du spectre, Fusion de données, Détection de brouilleurs, Communications tactiques, Réseaux ad hoc, Communications sans fil

Development of Spectrum Sensing Cooperation and Fusion Strategies for Tactical Heterogeneous Networks in Mobile Environments

Bryan GINGRAS

ABSTRACT

Spectrum sensing in tactical wireless networks that are under attack by jammers is an important consideration to ensure the safety and effectiveness of deployed military personnel. It is necessary for members of these networks to be aware of which channels in the spectrum are compromised and which are safe to use for data transmission. Wireless transmitters that compose these networks can accomplish this by sensing the energy level on different channels in order to determine if there are jammers active on these channels. They can then share this information with their peers in order to collaboratively identify and avoid jammers. Several solutions based on reinforcement learning exist that allow wireless transmitters to devise a transmission policy based on their observations of the jammers' activity, but these solutions often falter when the behaviour of a jammer is random, thereby preventing reinforcement learning algorithms from learning and anticipating their behaviour.

In this thesis, we first discuss collaborative spectrum sensing and the theory behind cognitive radios, jamming, and anti-jamming. Next, we detail the system model used to represent the multi-agent anti-jamming problem. We then introduce a collaborative pseudo-random channel selection algorithm and a data collaboration and fusion scheme based on super-decision vectors in order to improve awareness of spectrum utilization across the network. Simulation results show that this solution leads to higher rates of detected jammers as well as an increase in the number of transmissions occurring on unjammed channels.

Keywords: Collaborative Spectrum Sensing, Data Fusion, Jammer Detection, Tactical Communications, Ad Hoc Networks, Wireless Communications

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

Page

LIST OF ALGORITHMS

LIST OF ABREVIATIONS

AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
BS	Base Station
CR	Cognitive Radio
CRN	Cognitive Radio Network
CRTC	Canadian Radio-television and Telecommunications Commission
CSI	Channel State Information
CSS	Collaborative Spectrum Sensing
CTS	Clear-To-Send
DoS	Denial of Service
DSSS	Direct Sequence Spread Spectrum
EGC	Equal Gain Combining
FC	Fusion Centre
FCC	Federal Communications Commission
GUESS	Gossiping Updates for Efficient Spectrum Sensing
HMM	Hidden Markov Model
IED	Improvised Explosive Device
IEEE	Institute of Electrical and Electronics Engineers
JDR	Jammer Detection Ratio

JIEDDO	Joint IED Defeat Organization
LRT	Likelihood Ratio Test
MARL	Multi-Agent Reinforcement Learning
MMSI	Minimum Mean Square Estimation
QoS	Quality of Service
OSA	Opportunistic Spectrum Access
PDR	Packet Delivery Ratio
PU	Primary User
RL	Reinforcement Learning
RSS	Received Signal Strength
RTS	Request-To-Send
SDR	Software-Defined Radio
SNR	Signal to Noise Ratio
SU	Secondary User
TD	Temporal Difference
TSR	Transmission Success Rate
WN	Wireless Node
WRAN	Wireless Regional Area Network

LISTE OF SYMBOLS AND UNITS OF MEASUREMENTS

a	Non centrality parameter
a_t^i	Action taken by WN i at time step t
$\boldsymbol{a}_t^i$	Array of actions taken by WN i and its neighbours at time step t
A_i	Set of possible sensing actions that WN i can take
d	Distance between a transmitter and a receiver
dB	Power of a signal (decibels)
$\boldsymbol{d}_t^i$	Decision vector of WN i at time step t
$\boldsymbol{D}_{i,t}$	Set of possible decisions that WN i can take for each channel at time step t
$\boldsymbol{\delta}_t^i$	Super-decision vector of WN i at time step t
γ	Signal to noise ratio
ε_n	Exploration-exploitation trade-off constant of collaborative pseudo-random channel selection algorithm
J	Set of all jammers in the environment
Hz	Frequency of a signal (Hertz)
λ	Decision threshold
M	Set of all wireless nodes in the environment
N_{FB}	Number of channels in the spectrum
N_{WN}	Number of Wireless Nodes in the environment
$p_{d,m,AWGN}$	Detection probability when using AWGN channels, with diversity order m

$\bar{p}_{d,m,Ray}$	Average detection probability when using Rayleigh fading, with diversity order m
$p_{fa,m}$	Probability false alarm with a diversity order of m
$p_{k,00}$	Probability of jammer k remaining in the idle state from one time step to the next
$p_{k,11}$	Probability of jammer k remaining in the active state from one time step to the next
P_T	Transmission power
$s_t^{i,j}$	Occupancy of channel j as seen by WN i at time step t
$\boldsymbol{s}_t^i$	Occupancy of each channel as seen by WN i at time step t
$\boldsymbol{s}_t$	State of the environment, i.e. each channel's occupancy as seen by each channel at time step t
σ^2	Variance of the observed signal
τ_t^i	Observation made by WN i at time step t
t	Current time step of the simulation
T	Duration of the simulation given in time steps
W	Set of all channels in the spectrum
x_t^i	Outcome of WN i's transmission at time step t

INTRODUCTION

A lack of available frequencies in the spectrum is forcing industry, academia, and governmental authorities to explore new spectrum allocation paradigms in order to meet current and future spectrum demands caused by the endlessly growing number of connected devices. Since its emergence, cognitive radio (CR) has established itself as a viable solution to the problem of spectrum scarcity. Instead of limiting access to frequency bands strictly to licensed users, cognitive radio technology proposes to allow non-licensed users to access these frequencies as well, provided they do not interfere with the transmissions of the channels' licensees. These secondary users could stop transmitting on a channel as soon as a licensed primary user (PU) begins transmitting on the channel, or they could modulate their transmission power to a point where their transmissions do not cause interference with those of the primary user.

Collaborative spectrum sensing refers to a network of cognitive radios independently scanning the spectrum for available bandwidth and sharing their sensing information with their peers in order to increase the throughput across the entirety of the network. In addition to increasing the number of data points that each node possesses with respect to the occupancy of the spectrum, collaborative spectrum sensing (CSS) allows CRs to compensate for physical phenomena such as path loss, shadowing, and fading which might negatively affect the reliability of each node's observations (Roozgard *et al.* (2012)).

Combining these potentially unreliable local observations into a single reliable datum that can be used for decision-making is known as data fusion. Members of a cognitive radio network (CRN) can use data fusion to combine the observations that they have received using collaborative spectrum sensing into a set of decisions giving the occupancy of each channel in the spectrum. Nodes can then consult this decision set and identify vacant channels that they can use for the purpose of transmitting data.

Like many modern technologies, militaries around the world rapidly found a use for cognitive radios. Military collaborative spectrum sensing is mostly similar to civilian use, in the sense that in both, there is a network of cognitive radios that work together to find unused portions of bandwidth that can be used for transmitting data until an external user moves into this frequency and begins its own transmissions. However, there is an important distinction between the two: while civilian CSS is primarily concerned with secondary users opportunistically exploiting bandwidth left vacant by primary (licensed) users who are not necessarily concerned with the activity of secondary users, as long as they can use their licensed spectrum bands without being hindered by non-licensed users, military cognitive radios are actively competing for spectrum access against entities that seek to impede their transmissions. These users are called jammers, because their primary purpose is to prohibit access to the channel and prevent military CRs from accessing the spectrum and communicating with their peers.

We can draw the following analogy between civilian and military CSS: military CRs (analogous to secondary users) must avoid transmitting on channels that are occupied by hostile jammers (analogous to primary users). However, an important difference lies in the fact that while civilian CRs must take care not to cause interference in the licensed users' transmissions, military CRs that transmit on jammers' frequencies risk having their transmissions eavesdropped on or scrambled to the point where their intended receivers cannot decode the transmitted messages. This exposes the users of the CR to physical danger, as military CSS can take place during military operations that include armed combat. In the same vein, certain jamming technologies actively seek out which channels are being used by military CRs, in order to occupy these channels and block the CRs' transmissions. Tactical networks must therefore contend with adversarial jammers and seek to overcome them by employing spectrum sensing and data fusion techniques that take into account the strategies used by the jammers to compromise the transmissions made by members of these networks.

0.1 Problem Statement

Spectrum sensing in tactical wireless networks is used to equip the tactical radios that compose these networks with information regarding the occupancy of different channels while the spectrum is being accessed by opposing forces, i.e. hostile jammers in our case. Transmitting data on channels that are being used by these adversaries can have negative consequences. We therefore wish to develop techniques that allow members of these networks to detect when channels are unsafe for transmission.

However, it is not sufficient for only a single node in a tactical wireless network to be aware of spectrum utilization by hostile jammers. In order to ensure that their peers do not transmit on compromised channels, nodes that become aware of a jammer attacking a given channel need to share this sensing information with their neighbours in order to increase the security of transmissions across the network.

This project focuses on developing a channel selection algorithm that tactical communications systems, deployed on the field, can use to identify channels that are under attack by jammers. This allows the individual nodes to know which channels must not be used when broadcasting data. This algorithm is based on the observed activity of the jammers, as well as the sensing observations of neighbouring radios. Furthermore, this channel selection algorithm encourages members of the network to collaborate in order to mutually increase the reliability of their observations, since harsh conditions in the environment can result in the nodes' observations being incorrect and these probabilities of error can be reduced if two or more neighbouring nodes coordinate their sensing actions to simultaneously observe the same channel. Therefore, this algorithm will lead to nodes detecting a greater number of active jammers.

In addition to the channel selection algorithm, we introduce a data collaboration and fusion scheme using super-decision vectors that allows nodes to share their sensing information with

each other and combine them into a set of decisions pertaining to the occupancy of each channel. This allows each member of the network to better identify vacant channels, leading to an improved rate of unjammed transmissions.

0.2 Objective and Methodology

The primary goal of this dissertation is to present a solution that leads to members of a tactical wireless network acquiring an improved awareness of the usage of the spectrum by hostile jammers, in order to ensure that they do not transmit on channels that would result in their transmissions being intercepted or jammed to the point where they can no longer be decoded by their intended receivers. The proposed solution takes into account the unpredictable behaviour of hostile jammers, in order to ensure that we are not biasing the representation of the jammer in our favour. In other words, we model the jammers in such a way that their behaviour cannot be learned, predicted or reliably anticipated. In addition, our solution introduces a data collaboration and fusion scheme that allows members of tactical wireless networks to make better use of the sensing observations made by their peers.

In order to reach this goal, we do the following:

- We begin by describing the theory behind cognitive radios as well as the motivation that led to their development. We then turn our attention towards data fusion techniques, including centralized and distributed strategies. Next, we present a detailed review of jamming and anti-jamming techniques, before turning our focus towards collaborative spectrum sensing in the particular context of tactical wireless networks.

- Next, we give a detailed overview of our system model that we use to represent the multi-user anti-jamming problem, as well as the stochastic game that models the interactions between the members of the tactical wireless network, the jammers that are attacking the network, and the frequency spectrum on which both groups of entities are transmitting. We

then introduce our solution to allow members of the network to gain an improved awareness of the spectrum usage by the jammers, which leads to a higher rate of unjammed transmissions within the network.

- Finally, we perform simulations to test the effectiveness of our solution against an existing solution found in the literature that uses multi-agent reinforcement learning to build policies that determine which channels each node will sense. Specifically, we judge these solutions in terms of the number of times that jammers are detected as well as the number of transmissions that occur on unjammed channels. To establish a performance base line, we will also compare the effectiveness of these two solutions against actions that are chosen randomly.

0.3 Publications

The research presented in this dissertation has been accepted for publication in the proceedings of the IEEE International Conference on Communications (ICC 2020, Dublin) under the title "Collaborative Spectrum Sensing in Tactical Wireless Networks".

0.4 Dissertation Organization

This dissertation is structured as follows: the first chapter serves as an introduction to the theory and basic concepts relating to cognitive radios and collaborative spectrum sensing.

The second chapter introduces our system model and includes a detailed study of the proposed channel selection algorithm, and data fusion and collaboration schemes.

Finally, the third chapter presents the simulation methodology, parameters, and finally the results obtained using our solution.

Figure 0.1 provides a graphical representation of the content of this dissertation.

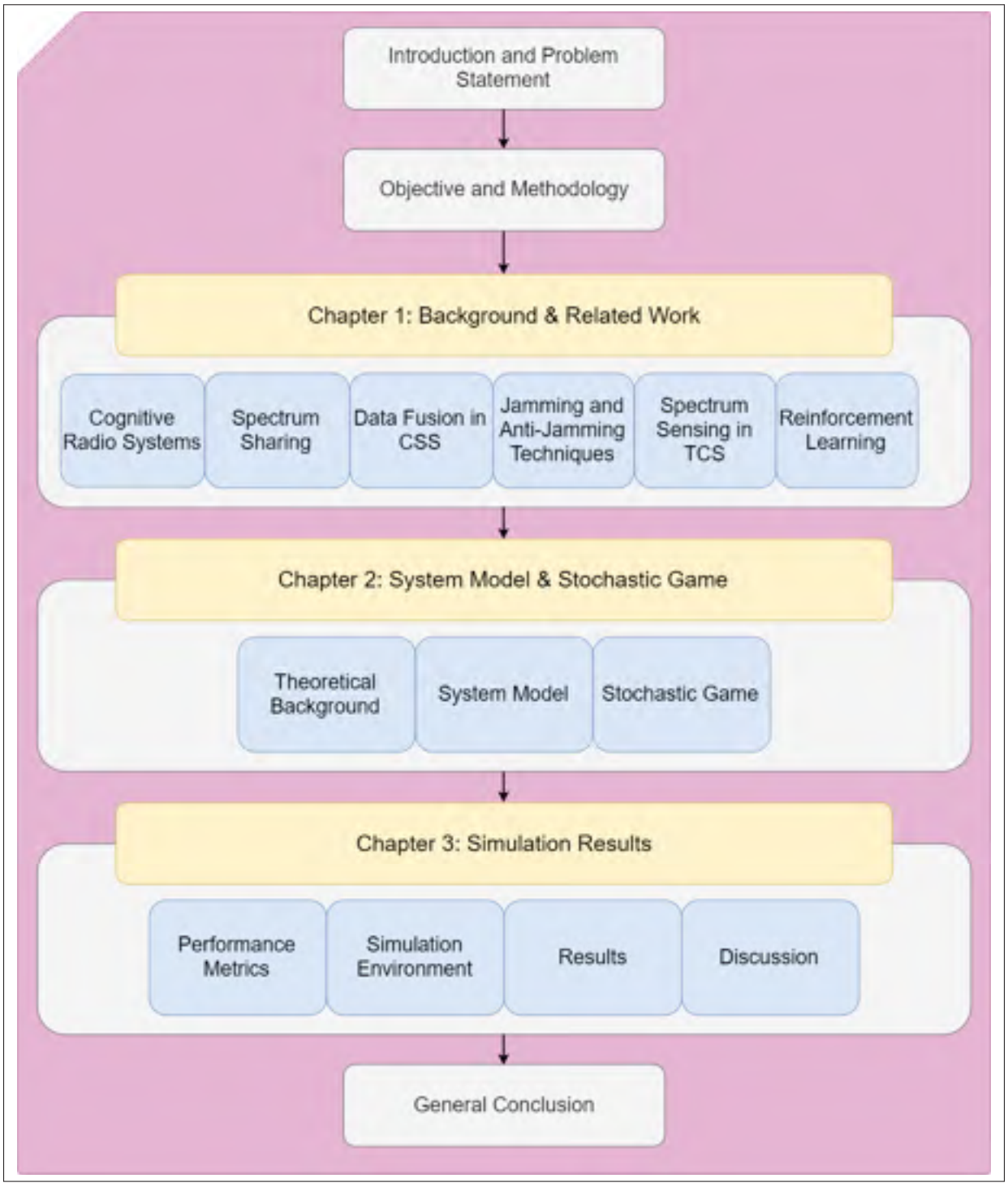


Figure 0.1 Chapters diagram

CHAPTER 1

BACKGROUND AND RELATED WORK

1.1 Introduction

Currently, access to the electromagnetic spectrum is partitioned and reserved for different categories of users by entities such as the United States of America's Federal Communications Commission (FCC) or the Canadian Radio-television and Telecommunications Commission (CRTC). The exclusive nature of spectrum allocation means that users who are not specifically licensed for a specific band of the spectrum are not entitled to communicate over the band in question. This rigidity is causing authorities to struggle to meet the spectrum demand driven by the ever-growing number of connected devices, which is believed to reach nearly 30 billion before the year 2022 (Ericsson (2017)).

This massive increase in spectrum demand drives a need for innovation with regards to spectrum allocation and usage. In fact, as early as 2002, the FCC's spectrum policy task force stated in a report that a spectrum policy reform was needed in order to continue to meet the public's need for spectrum access. Furthermore, the report indicates that many large portions of the licensed spectrum are frequently unused for large periods of time, resulting in a massively inefficient usage of the spectrum, all while dealing with spectrum shortages. The report states that an updated policy should allow increased flexibility of spectrum use by licensed and unlicensed users (Kolodzy (2002)). This would mean that although licensed users would maintain priority access on their portion of the spectrum, secondary (unlicensed) users would be permitted to access the spectrum while it is not in use by the primary (licensed) user, thereby offering a solution to the problem of licensed spectrum remaining vacant for long periods of time as well as making progress towards meeting the demand of the growing number of connected devices.

Before a secondary user (SU) can exploit bandwidth reserved to a primary user, it must first observe and analyze the signals present on a given channel in the spectrum in order to determine

if the channel is vacant or if it is being used by another transmitter. If the detected power level is sufficiently low, the SU can then opportunistically access the spectrum and use it for transmission until the PU becomes active and seeks to use the spectrum to which it is entitled. Secondary users and their peers can increase their overall knowledge of the spectrum (and therefore, their opportunities to access the medium) by sharing their respective observations of the spectrum with each other, giving rise to a field of research known as collaborative spectrum sensing (Ghasemi & Sousa (2005)).

Governments are investing considerable resources into researching spectrum collaboration to address the problem of spectrum shortage. In 2016, the US Department of Defense launched the Spectrum Collaboration Challenge (SC2) to encourage academics and professionals to develop collaborative spectrum access techniques that make use of the most recent advances in artificial intelligence (AI) and software-defined radios (SDR), with the ultimate goal of replacing the current spectrum allocation policy with one that emphasizes opportunistic access to the spectrum (Tilghman (2016)).

1.2 Cognitive Radio Systems

Devices that scan the network to find available bandwidth are known as cognitive radios. These can communicate with each other to form cognitive radio networks. Aside from spectrum sensing, important properties of CRs are spectrum mobility (seamlessly switching to an unused channel if the PU becomes active again), as well as their ability to intelligently choose, after detecting multiple unused channels in the spectrum, which is the best to use for transmission, considering different parameters such as required transmission power, modulation schemes, and its own QoS (quality of service) requirements (Kusaladharma & Tellambura (1999)).

1.2.1 Cognitive Radio Paradigms

With the evolution of cognitive radio technology, the definition of a cognitive radio system has broadened over time. Today, the functioning of CRs can be classified into three separate mod-

els: the underlay, overlay, and interweave paradigms. The underlay paradigm assumes that the SUs have knowledge of the interference that their transmissions would incur on a PU should they transmit simultaneously on the same channel (Awan *et al.* (2009)). They must therefore limit their transmission power in order to avoid interfering with the PU's transmissions. Techniques that can be used to reach the power threshold include spread spectrum, where the SU spreads its message across a large frequency band with power below the noise floor. The SU's receiver can then reverse the spread and decode the transmission.

The overlay paradigm differs from the underlay as it assumes that the SUs possess knowledge of the PU and its transmissions. An SU can use this knowledge to mitigate the interference perceived by the PU caused by the SU transmitting on its channel. Due to its awareness of the PU's transmissions, a secondary user can use some of its transmission power to relay the PU's broadcast and increase its signal to noise ratio (SNR). Careful selection of the transmission power can even allow the SU to compensate for the SNR drop it causes to the PU by allocating just enough power to relaying the PU's transmission to increase the SNR by the same amount by which it lowers it (Wang (2009)), thus ensuring that licensed and unlicensed users can coexist without interfering with each other.

The third CR model is the interweave paradigm, which is based on opportunistic spectrum access (OSA), where SUs sense the spectrum and occupy channels temporarily left vacant by licensed users. It is generally assumed with this paradigm that it is unacceptable to have SUs and PUs simultaneously occupying the same channel. In other words, the SU can utilize a channel for as long as it is not being actively used by the licensed PU. If and when the PU begins accessing this channel again, the SU must desist and find a different channel on which it may transmit. The three CR paradigms are summarized in Figure 1.1.

1.2.2 Physical Layer Sensing

The different possible detection methods that CRs can use are divided into three categories: non-coherent detection, coherent detection, and feature detection (Sahai *et al.* (2004)). Among

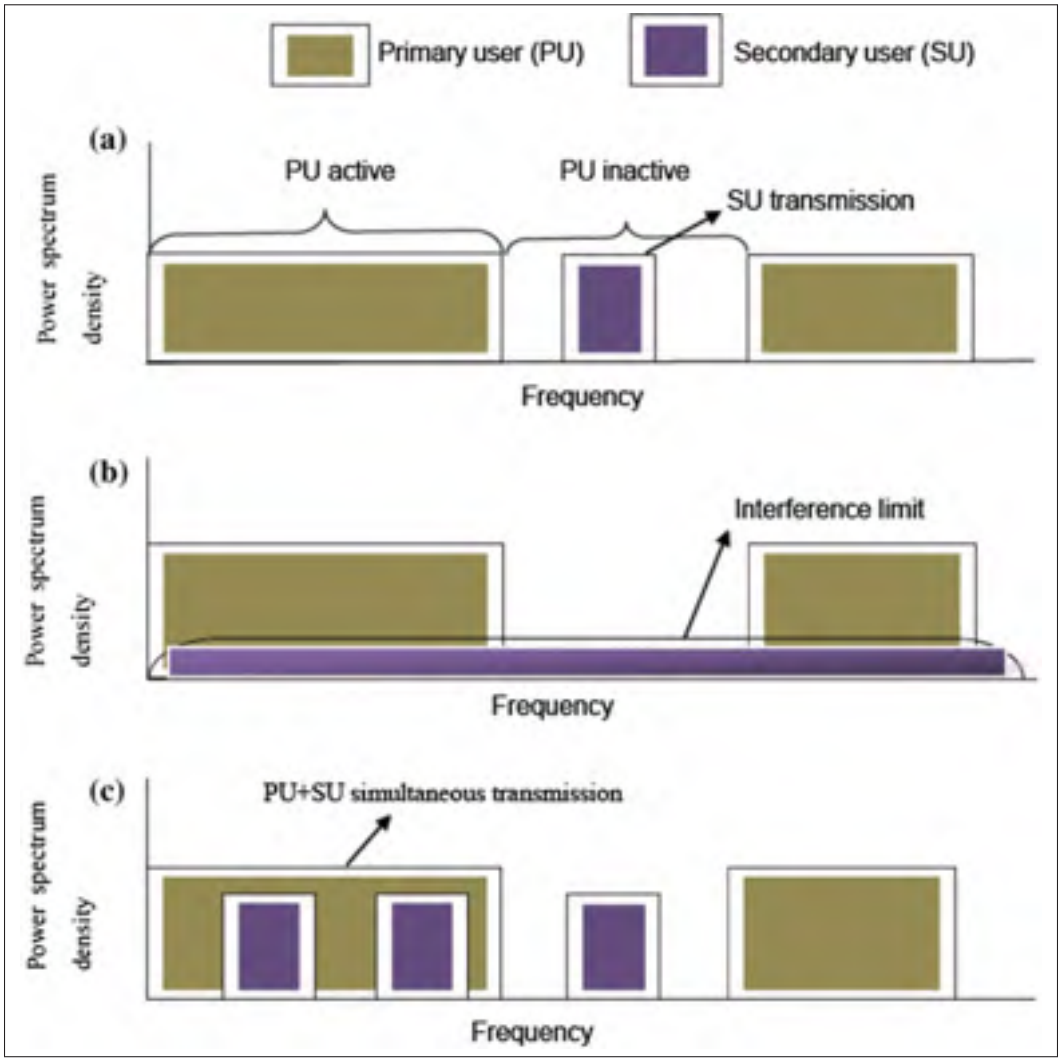


Figure 1.1 Cognitive radio paradigms: (a) interweave, (b) underlay, (c) overlay
Taken from Pandit & Singh (2017)

these, the simplest and most common method is the non-coherent energy detection method, which consists of reading the power level of a given frequency channel. This method is popular due to its low sensing time requirement as well as the fact that it does not require any prior knowledge of the signal. However, this technique requires the strength of the signal to be greater than the ambient noise, otherwise the CR will be unable to detect the signal (Wang (2009)).

Matched filter detection is a well known coherent detection method that can be used when the CR possesses some knowledge of the signal it is attempting to detect. In matched filter detection, the CR uses its *a priori* knowledge of the signal as the filter, which it convolves with the received signal. A maximal SNR of the resulting signal indicates that the two are a match. Although matched filter detection outperforms energy detection in low SNR environments (Sahai *et al.* (2004)), its implementation in practice is limited due to its high complexity and timing concerns.

Feature detection is a method that functions similarly to matched filter detection, since it also matches known properties of the signal (in this case, cyclostationary properties) with the observed signals. However, much like energy detection, its use is conditional on the signal being sufficiently strong relative to noise.

1.2.3 IEEE 802.22 Standard

In 2009, the IEEE, in collaboration with the FCC, developed the IEEE 802.22 WRAN standard, which is the first attempt at standardizing technologies that allow opportunistic use of unused licensed spectrum. This standard was developed in order to bring broadband Internet access to sparsely populated rural areas by exploiting frequencies licensed to television broadcasters who underutilize their assigned bandwidth (Stevenson *et al.* (2009)). CR devices using this standard can find vacant channels in the spectrum using two methods: energy detection as well as geolocation and a database. This second method requires the SU and the base station (BS) to be aware of each other's geographical location. Using this information, the BS can consult its database, which is a comprehensive summary of protected television broadcasting information in the area, and provide each SU with the channels on which it can transmit as well as the maximum transmission power that it can use without causing interference in the licensee's broadcast.

In 2010, the IEEE approved a complementary standard to IEEE 802.22, called IEEE 802.22.1. The motivation behind this second standard is to offer greater interference protection for tele-

vision broadcasters who use low-power wireless devices such as wireless microphones. These devices use the same licensed frequencies as high-power television broadcasts, which makes them far more susceptible to interference from SUs transmitting on their licensed regions of the spectrum (IEEE (2010)).

1.3 Spectrum Sharing

When using opportunistic spectrum access, the interweave paradigm assumes that access to a channel is mutually exclusive between a PU and an SU. As we saw with the overlay and underlay paradigms, it is possible for the SU and PU to use simultaneously use the same channel in the spectrum, although the former must take care to limit its transmission power to avoid causing interference in the latter's transmissions. At the same time, the SU's transmission power must be high enough to overcome ambient noise and signal fading. Figure 1.2 shows a PU and an SU sharing a channel, where the SU is limiting its transmission power to below a threshold, beyond which it would interfere with the PU. Power allocation is therefore an important consideration in situations where SUs and PUs can share the spectrum. We consider different power allocation strategies depending on whether we represent the CRs as having perfect or imperfect channel state information (CSI). In other words, power allocation strategies used by the SUs to limit their transmission power depends on whether or not they are aware of statistical information such as the channel's fading distribution, channel gain, etc.

1.3.1 Secondary User Capacity with Perfect State Information

Optimal power allocation strategies for CRs with perfect channel state information are presented in Kang *et al.* (2009). The article demonstrates the peak and average power constraints to which the SU must adhere in order to ensure that the QoS of the PU is unaffected. These constraints are based on factors including the gain of the channel used for communication between the SU and the PU, the peak transmission power limit of the PU, and the peak received power at the PU. Additionally, the article also demonstrates that the gain of the PU-SU channel is inversely proportional to the SU capacity, up to a certain threshold given by the PU's trans-

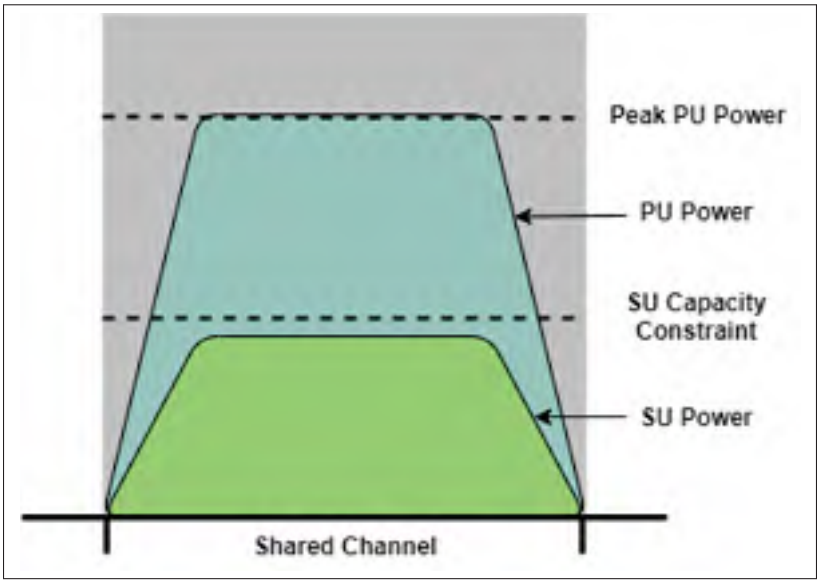


Figure 1.2 SU and PU sharing a channel

mitted and received power levels. In other words, channel fading between the SU and PU can have a positive impact on the SU's transmission capacity.

1.3.2 Secondary User Capacity with Imperfect State Information

Since it is often infeasible to obtain perfect CSI, research has been conducted to determine the SU's capacity when performing spectrum sharing with imperfect information on the channel. In such cases, it becomes necessary to estimate the channel state information that the SU is missing. In Musavian *et al.* (2009), the authors assume that the channel gain experienced at the PU's receiver is unknown to the SU, which motivates the SU to perform minimum mean square estimation (MMSI) to approximate this gain using information fed to it either by a band manager or directly from the PU's receiver. In either case, the information the SU receives is not a perfect representation of the channel gain, which is the reason it is called imperfect CSI. It is then possible for the SU to determine its transmission capacity as a function of the PU's transmission power as well as the known and estimated channel gains.

Son *et al.* (2013) compares the SU's capacity for both perfect and imperfect CSI, and shows that the SU's capacity in the latter case is less than half of what could be achieved with perfect CSI. This emphasizes the importance of knowing, or at least estimating to a high degree of confidence, the CSI in order to ensure a higher capacity for the SU's transmissions when sharing the spectrum with a PU. The article further demonstrates the impact of having multiple SUs or PUs simultaneously sharing the spectrum: as the number of SUs increases, multi-user diversity gain results in higher capacity for each secondary user, while an increase in the number of PUs imposes a tighter restriction on the SUs' transmission power.

1.4 Data Fusion in Collaborative Spectrum Sensing

Considerable gains in sensing accuracy are possible when the data independently gathered from multiple sensors regarding the same channel are combined, using one of several existing data fusion strategies, into a singular datum. In the case of collaborative spectrum sensing, cognitive radios that are geographically near each other can form a CRN and share spectrum sensing information, allowing each of them to gain a more accurate understanding of the usage of the spectrum by its licensed users. This gain in information improves the ability of each member of the CRN to find unused bandwidth that can be used for transmission (assuming that we are using the interweave paradigm which forbids sharing the spectrum with PUs).

Two major CSS data fusion schemes exist: centralized and distributed data fusion. Both have important constraints with regards to reliability, transmission overhead, and the requirement for the presence of existing telecommunication infrastructure. These are called centralized and decentralized (also known as distributed) data fusion.

1.4.1 Centralized Data Fusion

Centralized data fusion in CSS represents the case where each member of the CRN transmits its spectrum observations (either an SNR reading or a local decision on whether or not the observed channel is jammed) to a fusion centre (FC), which is responsible for combining all

received observations into a set of decisions relative to the occupancy of each channel in the network. It then broadcasts these decisions back to the SUs, who then consult them and select a vacant channel for transmission (Akyildiz *et al.* (2006)). Figure 1.3 gives a representation of centralized data fusion.

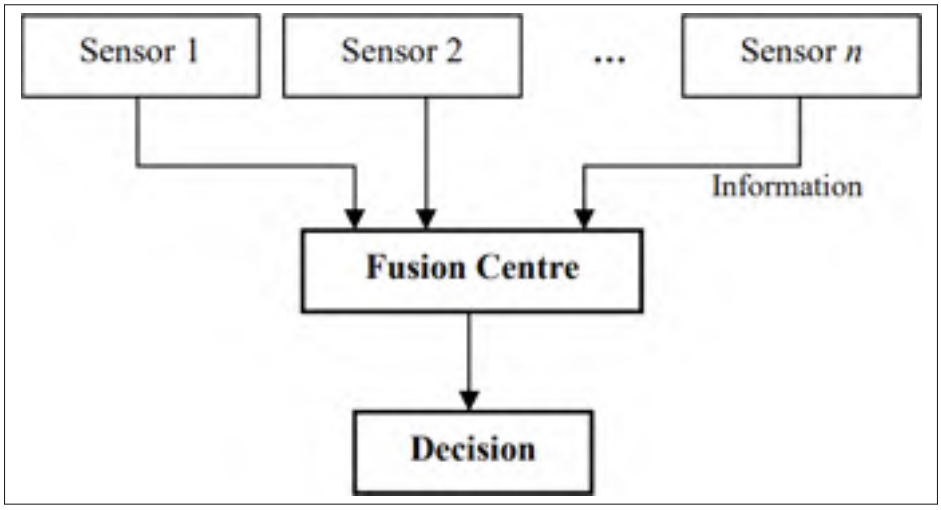


Figure 1.3 Representation of centralized data fusion
Taken from Vesolo (2009)

One of the major hindrances of centralized data fusion is its reliance on existing infrastructure to fill the role of fusion centre. This means that centralized data fusion is significantly more difficult, if not impossible, to apply in deserted environments or areas such as disaster zones where potential FCs are damaged or destroyed (Lien *et al.* (2009)). Another important limitation to consider is the added traffic overhead caused by each CR broadcasting its sensing information to the FC. In fact, the cost of this overhead is often prohibitive, meaning that it is not feasible for each SU to transmit its entire observation to the FC (Bhattacharya & Saha (2014)).

1.4.1.1 Hard-Decision Making

A simple solution to the overhead problem is for each SU to make its own decision on whether or not the channel it has observed is vacant. In other words, the responsibility of analyzing sensing observations to determine occupancy is offloaded from the FC to the SUs. After each

secondary user has interpreted its observation, it only needs to send a single bit of information to the FC indicating its local decision, e.g. 1 for occupied, 0 for vacant. After receiving each node's one-bit decision, the FC applies a data fusion rule and renders a final decision regarding the occupancy of each channel (Visotsky *et al.* (2005)).

There exist several methods for combining sensing data into a single decision, the most popular being the AND-rule, OR-rule, and Majority-rule. In the AND-rule, the FC will judge a channel to be occupied if all the local decisions pertaining to that channel indicate that the channel is occupied. Conversely, it will judge the channel to be vacant if a single local decision infers that it is vacant, even if all the others allege that it is occupied. On the other hand, if the FC applies the OR-rule, it will judge a channel to be occupied if a single local decision shows it to be occupied. The Majority-rule, as its name suggests, adheres to the local decision that is most often reported by the SUs (Cai *et al.* (2014)). The simplicity of these fusion rules results in very low temporal and computational overhead, making them ideal to use in environments with scarce resources.

1.4.1.2 Soft-Decision Making

If bandwidth limitations are not a concern, then soft decisions are the ideal choice. In soft decision making, each SU sends its entire observation (i.e. the SNR it measured on a given channel) to the PU, who then combines these observations into a set of decisions. It has been demonstrated that the Neyman-Pearson Lemma (Visotsky *et al.* (2005)), applied as a likelihood ratio test (LRT) on the vector of received SNR readings, is the optimal soft information combining algorithm. It is also demonstrated that if there was no cost associated with the added traffic overhead, then centralized data fusion would be the optimal fusion strategy (Castanedo (2013)).

1.4.1.3 Data Quantization

Data quantization is an attempt to achieve a performance close to that of soft-decision making while limiting the prohibitive traffic overhead that comes from using this technique. Instead of sending their entire observations, SUs instead compress their sensing data into a multibit quantized format which is then sent to the FC, which then applies a soft information combining algorithm such as equal gain combining (EGC) or likelihood ratio test. Naturally, this quantized datum generates less traffic overhead than an entire SNR observation. Fu *et al.* (2017) demonstrates that quantizing SUs' observations to as few as 5 or 6 bits yields performance nearly identical to what is achieved using soft fusion, which makes it an attractive alternative to sending the entire observation.

1.4.2 Distributed Data Fusion

When the environment in which the CRN operates does not contain any wireless infrastructure that can serve as a fusion centre, the members of the network have no choice but to fulfill the FC's tasks themselves. This means that each SU will sense the spectrum, decide whether or not the sensed channel is occupied by a licensed user, and share this conclusion with the nodes nearest to it. This is represented below in Figure 1.4.

Decentralized data fusion possesses another advantage: since an SU that is geographically further away from a PU is less likely to detect the PU's signal (as it will have attenuated over distance), sensing information that it receives from one of its peers that is closer to the PU will be more likely to be correct since the nearer SU will perceive a stronger signal than nodes that are further away from the PU. In other words, CRs that are closer to the PU can assist those that are further away, thus increasing each node's awareness of the spectrum and therefore its chances of accessing the spectrum (Yucek & Arslan (2009)).

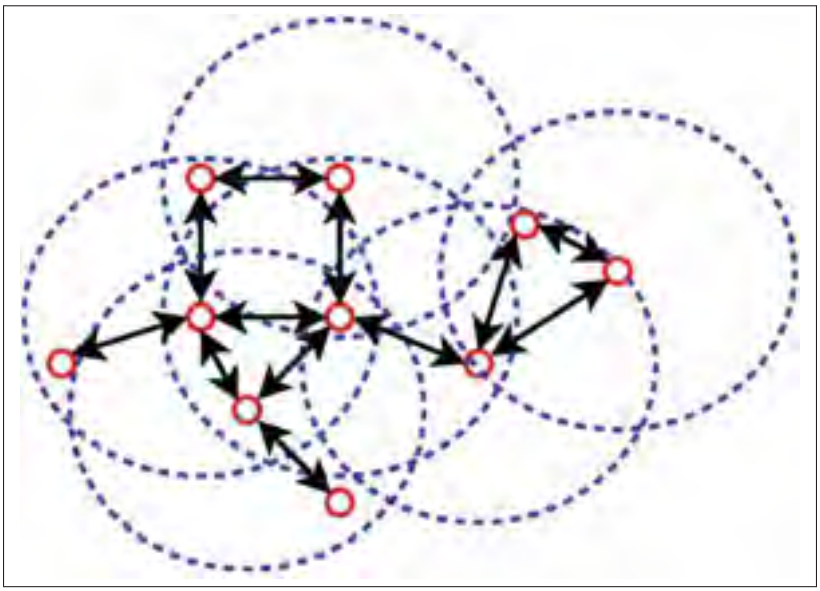


Figure 1.4 Representation of how sensing observations are shared when employing distributed data fusion Taken from Lundén *et al.* (2013)

1.4.2.1 Inter-Node Communication

Among the existing literature that studies the efficient communication of data between neighbouring nodes is the GUESS (gossiping updates for efficient spectrum sensing) technique proposed by Ahmed *et al.* (2006). Essentially, when a node has information to send, it will randomly choose one of its neighbours and share its sensing information with it. Once the neighbour has received this information, it too chooses a random neighbour and repeats the process. This continues until all nodes have received the information. It was shown that this solution scales according to $O(log\ n)$ where n is the number of nodes in the network.

1.4.2.2 Scheduling the Sharing of Sensing Information

Literature in the field of CSS generally assumes that inter-node communication occurs on a control channel that can never be occupied by a PU. While this means that the ability of the SUs to communicate with each other is never compromised, it adds a new challenge due to the fact that if a node simultaneously receives sensing information from two different neighbours

(an example of the hidden terminal problem), both transmissions will interfere with each other and the node will be unable to decode either observation. Lundén *et al.* (2015) proposes a scheduling algorithm that is guaranteed to rapidly converge to a schedule that ensures collision-free sharing of spectrum sensing information, if such a schedule is possible given the layout of the network.

1.5 Jamming Algorithms and Jamming Mitigation Techniques

Ever since the technology's inception, telecommunications have been used by military forces around the world. Whether it is for tactical purposes, such as a platoon of soldiers communicating with each other on the field, or for larger-scale purposes such as between fleets of ships in the Pacific Ocean, radio communications are an integral part of how today's armed forces conduct operations.

Radio technology is also used by armed insurgent groups, such as the ones in Afghanistan and Iraq, which can use wireless communication devices such as cellular or satellite phones to trigger improvised explosive devices (IEDs). Such devices were responsible for over 70% of American combat casualties (dead or wounded) in Iraq and over 50% in Afghanistan (Wilson (2007)). In response to these events, the US Department of Defense has established the Joint IED Defeat Organization (JIEDDO), which collaborates with different stakeholders to devise IED countermeasures. Among the many different solutions that this organization and its partners developed is the IED Countermeasures Equipment (ICE), which is a sophisticated, vehicle-mounted jamming device that identifies and jams radio frequencies used to trigger IEDs (Daily (2005)). These devices are not perfect, as they can also affect friendly transmissions, and insurgents were quick to develop counter-countermeasures (Wilson (2007)), effectively locking both sides into an endless arms race. Jamming and anti-jamming techniques are therefore an important topic of study when discussing collaborative spectrum sensing, even in a non-military context, as jammers can threaten the functioning of a network and cause a denial of service (DoS).

1.5.1 Jamming Algorithms

Jammers can use a variety of different techniques to affect transmissions within the network they are attacking. They vary in terms of their energy efficiency, their effectiveness, their complexity, as well as their ability to jam multiple channels. Broadly speaking, jamming algorithms are divided into two large categories: elementary and advanced. Elementary jammers can be further categorized as proactive or reactive, and advanced jammers can be described as function-specific or smart-hybrid (Grover *et al.* (2014)). Most of these techniques focus on the physical layer, but cross-layer algorithms exist as well.

1.5.1.1 Elementary Jammers

Proactive jammers, like their name suggests, operate without consideration for any user transmitting on the channel. This type of jammer does not switch channels, and transmits until its energy is exhausted. The simplest implementation of this approach is for the jammer to transmit continuously on a single channel, but this type of jamming is easy to detect and very inefficient in terms of energy consumption. A second implementation of proactive jamming is to occasionally switch between active and idle states in order to conserve energy. This results in a trade-off between the effectiveness of the jammer and its energy consumption. The time that a jammer spends in either state is adjustable, in order to strike the desired balance of the trade-off (Xu *et al.* (2005)).

Reactive jammers, on the other hand, only become active when they detect transmissions occurring on their assigned channel. Although they are much more difficult to detect than proactive jammers, they are much less energy efficient than other techniques such as random jamming due to having to constantly monitor the network for transmissions to jam (Grover *et al.* (2014)). An implementation of this technique consists of jamming the channel after a transmitter emits a request-to-send (RTS) message, in order to prevent the intended receiver from successfully responding with a clear-to-send (CTS) message. Since the transmitter does not receive a CTS in response to its RTS, it believes that the receiver is busy with another transmitter and will

not transmit data. Similarly, the jammer can allow the receiver to send the CTS message, and jam the channel after the node receives the transmission, thus preventing it from responding with the expected ACK packet. The transmitter, believing that the receiver did not receive the message, is forced to resend its data (Pelechrinis *et al.* (2010)).

1.5.1.2 Advanced Jammers

Function-specific jammers are proactive jammers that follow a predetermined strategy for choosing which channels they will attack, along with the power that they will allocate to each of them. This approach offers greater flexibility in terms of both energy efficiency and jamming effectiveness, as well as being less predictable than algorithms that focus only on a single channel. The follow-on algorithm is an example of function-specific jamming, and consists of rapidly switching between all channels using a pseudo-random frequency hopping sequence (Grover *et al.* (2014)). The channel-hopping jamming algorithm also uses a pseudo-random hopping code to select which channels it will attack at any given moment using a subset of all channels. Pulsed-noise jammers can also attack multiple channels both by spreading their power over a larger band of frequencies and switching from one group of channels to another (Muraleedharan & Osadciw (2006)).

Smart-hybrid jammers attempt to maximize their impact on the network while minimizing their energy consumption, which is why they are called smart. The reason they are called hybrid is because they can be both proactive and reactive. For example, control channel jammers seek to compromise a node in the network, which gives the jammers access to the network's control channel, potentially halting the operation of the entire network. If the network attempts to reconfigure and select a new control channel, this information can be sent to the compromised node, which will then jam the new control channel, and so on (Lazos *et al.* (2009)). Another example of a smart-hybrid approach is implicit jamming, which degrades the performance of the entire network by attacking a node's rate adaptation algorithm, forcing the BS to lower its transmission rate when servicing that node and reducing the amount of time available to address the other nodes in the network (Broustis *et al.* (2009)).

1.5.2 Anti-Jamming Algorithms

Anti-jamming techniques have been developed in order to counteract the jamming algorithms described in the previous section. Anti-jamming consists of jammer detection, i.e. becoming aware of which channels are under threat by jammers and avoiding transmission on these channels, as well as jamming countermeasures, which means recuperating from a collision with a jammer.

A simple way to recover from a jamming attack is to physically distance oneself from the jammer in hopes of exiting its transmission range. This can change the topology of the network, which means that nodes must communicate their coordinates to each other in order to determine the new configuration of the network (Grover *et al.* (2014)). In the case of networks with a BS, handoff from one BS to another may be necessary if the node travels too far from its original access point.

Another simple and popular anti-jamming technique is proactive channel hopping, which consists of switching from one channel to another after a set period of time has elapsed, regardless of whether the channel is jammed or not. This strategy is effective when the network has access to a large number of orthogonal channels, otherwise the energy spill-over on channels adjacent to a jammed channel can render channels unavailable as well, even if they are not directly being jammed. Broustis *et al.* (2009) demonstrate that a network with 12 orthogonal channels can be completely jammed by as little as four equidistant jammers, since each jammer disables not only the channel on which it is broadcasting, but also the ones adjacent to it due to energy spill.

An example of a more sophisticated anti-jamming technique is the Hermes node (Mpitziopoulos *et al.* (2007)), which uses a combination of direct sequence spread spectrum (DSSS) and frequency hopping spread spectrum (FHSS) to avoid follow-on jammers. Hermes nodes switch channels very rapidly (one million times per second) across 55 channels, and spread their signal over 275 MHz using a secret code that is known by all members of the network.

In terms of jammer detection, it is possible to use alternative strategies when conventional techniques such as energy detection are not sufficient. One such approach is detecting reactive jamming using the bit error rate (BER) of received packets (Strasser *et al.* (2010)). If the received signal strength (RSS) is low, and the bit error rate is high, then one could reasonably conclude that errors in the packet are due to noise. On the other hand, if there are bit errors while the RSS is high, then we can conclude that there is an active jammer on the channel. This jammer detection approach works well in the short term, unlike others such as using a packet delivery ratio (PDR) threshold which need a longer period of time before concluding if a channel is jammed.

1.6 Spectrum Sensing and Sharing in Tactical Wireless Networks

SDR technology has grown tremendously over the years. This technology allows its users to share the spectrum with other radios in the network by intelligently selecting transmission frequencies, spreading the transmission over a larger frequency band, and by correcting errors that occur due to path loss and signal fading (Staple & Werbach (2004)). SDRs also use cognitive radio technology to coordinate spectrum usage with other nodes in the network in order to avoid mutual interference. For these reasons, SDR technology has become an integral communication tool for military forces around the world, generally taking the shape of handheld, manpack or vehicle-mounted tactical communications systems. While the amount of wireless data needed to ensure the effectiveness of deployed military personnel is greater than ever, the amount of spectrum available for transmission is often highly limited (Harris (2019)). Equipping military personnel with cognitive radio systems that are able to intelligently sense the spectrum for available bandwidth and also share this information with their peers is therefore of the highest priority.

In addition to collaboratively sensing the spectrum for available channels, networks of tactical communications systems must contend with hostile jammers that seek to block or intercept transmissions between members of the network, which can jeopardize the safety of soldiers on the battlefield by isolating them from their peers. An example of such a device is Sen *et al.*

(2012), which is a patent for an ad hoc network of devices that can detect wireless communications and corrupt, spoof or jam them. If a plurality of these devices exists in the network, i.e. if they outnumber their adversary's transmitters, they can achieve network dominance and prevent any hostiles from transmitting. Figure 1.5 shows the overall operation of this jammer.

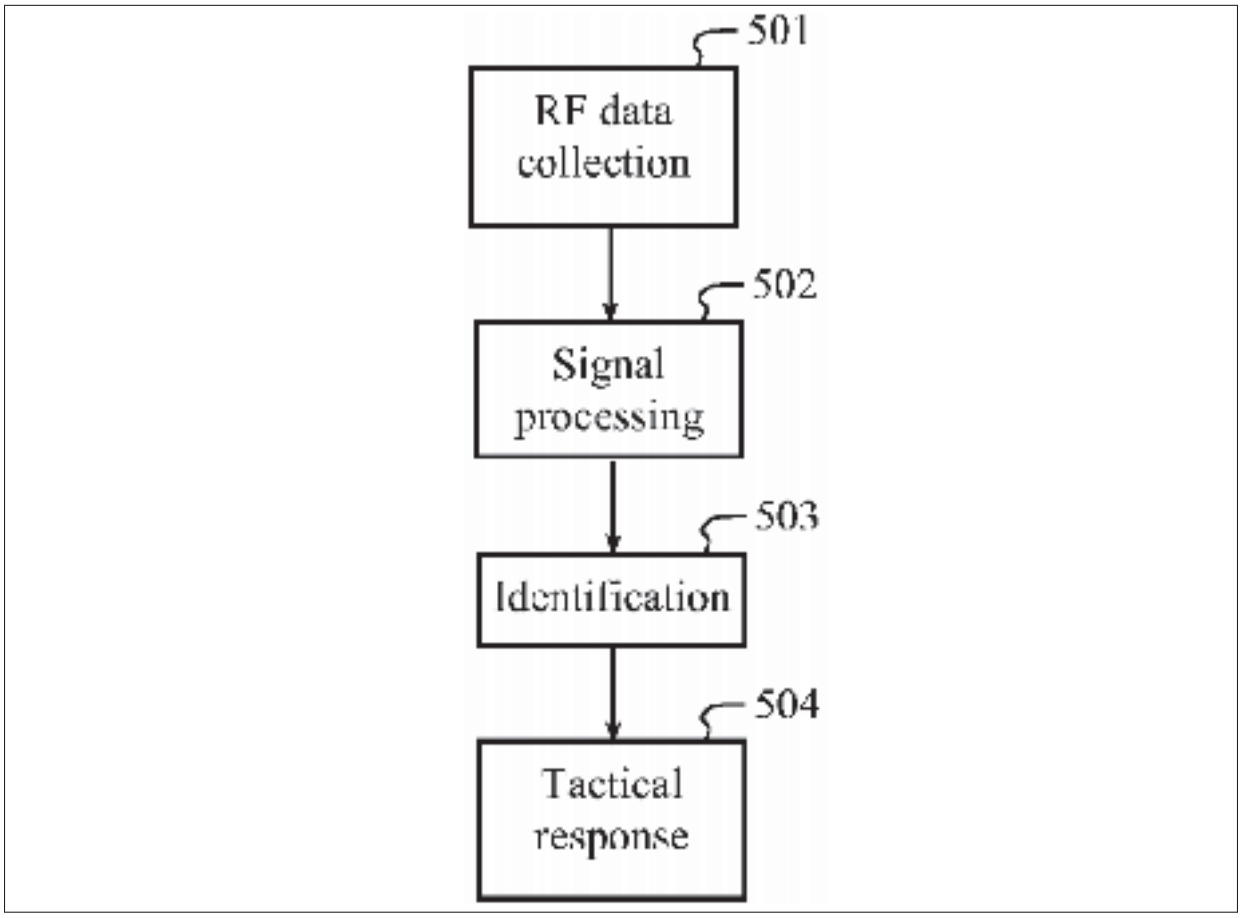


Figure 1.5 Overview of military jammer operation
Taken from Sen *et al.* (2012)

Military anti-jamming requires specialized solutions adapted to the tactical context. Seo *et al.* (2011), for example, describe a beam-steering technique that can be implemented on SDRs in order to prevent GPS signals (which are particularly vulnerable to jammers due to their low signal strength) from being jammed. It does so by tuning the radio's antenna to the signals of satellites and filtering out other transmissions such as those originating from hostile jammers. Although this solution is successful in limiting the effects of jammers or interference on GPS

signals, it requires that users be immobile, which limits the number of situations in which it can be used.

The presence of jammers means that cognitive radio networks used in a military context are faced with an additional difficulty that civilian applications generally do not have to face. This means that research in the field of tactical wireless networks must jointly consider the CSS and anti-jamming problems. Furthermore, the quantity of available research that simultaneously addresses both of these problems is severely lacking, which is one of the motivations for the research project described in this dissertation.

In addition to the presence of jammers, members of tactical wireless networks must respect the fact that other entities in the environment may have priority access to the spectrum, much like in a civilian context. For example, in Wang *et al.* (2013), the authors represent a scenario where soldiers equipped with portable radios act as PUs and wireless sensors act as the SUs. As the soldiers move across the terrain, they enter the transmission range of various sensors, who must then cease transmitting in order to ensure that they do not interfere with the soldiers' transmissions. In order to determine whether or not any soldiers have entered their transmission range, the SUs share their sensing observations with neighbours with whom they are correlated. Using this belief propagation technique, the SUs are able to disseminate their observations to their neighbours while reducing the effect of noise and fading on the nodes' individual observations. Results show that this scheme leads to faster detection of the PUs by the SUs, at the cost of greatly increased traffic in the network. Therefore, an important consideration in this field of study is to develop techniques that improve SUs' knowledge of when PUs are active, while limiting transmission overheads generated by these techniques. Although it is essentially impossible for SUs to cooperatively increase their knowledge of the spectrum without creating more traffic in the network, the benefit gained by this increased knowledge may outweigh its costs.

1.7 Reinforcement Learning

A popular tool today for a wide variety of Engineering problems is Reinforcement Learning (RL), which is a branch of Machine Learning that is concerned with the interactions between software agents and the environment in which they exist. Through interacting with their environment, RL agents learn to choose their actions in order to maximize a reward quantity that their actions generate. For example, RL can be applied towards playing chess (Silver *et al.* (2018)), where the state can consist of the position of all the chess pieces on the board, and the actions consist of the possible moves that the player can make at that time.

Figure 1.6 shows how RL agents interact with the environment. Based on the current state of the environment, the agent performs an action A_t which yields a reward R_{t+1}. The action may also alter the state of the environment, causing it to enter a new state S_{t+1}. Knowing that it obtained a reward of value R_{t+1} by taking action A_t while in state S_t, the agent can assume that taking the same action while in the same state would yield the same reward. Therefore, by keeping a tally of the rewards gained by taking different actions while in different states, the agent can learn which actions lead to the most reward in each state S_t. Over time, agents can converge to a situation where they know which action to take when they are in each state in order to maximize their reward. This state-to-action mapping is called a policy, represented by π. A policy π is said to be optimal if its total expected reward is greater or equal to any other possible policy for all possible states. Optimal policies are denoted using π_* (Sutton & Barto (2018)).

Agents seek to maximize their total discounted return G_t, i.e. to select their next action, they will not only consider the reward of that action, but also the future rewards gained by taking future actions. The hyperparameter γ is called the discount factor, and it determines the importance of future rewards. If $\gamma = 0$, the agent is myopic, meaning that it would only consider the potential reward of the next action. Conversely, if $\gamma = 1$, the agent would apply as much consideration to all of the future actions as it would the next action. The total discounted return

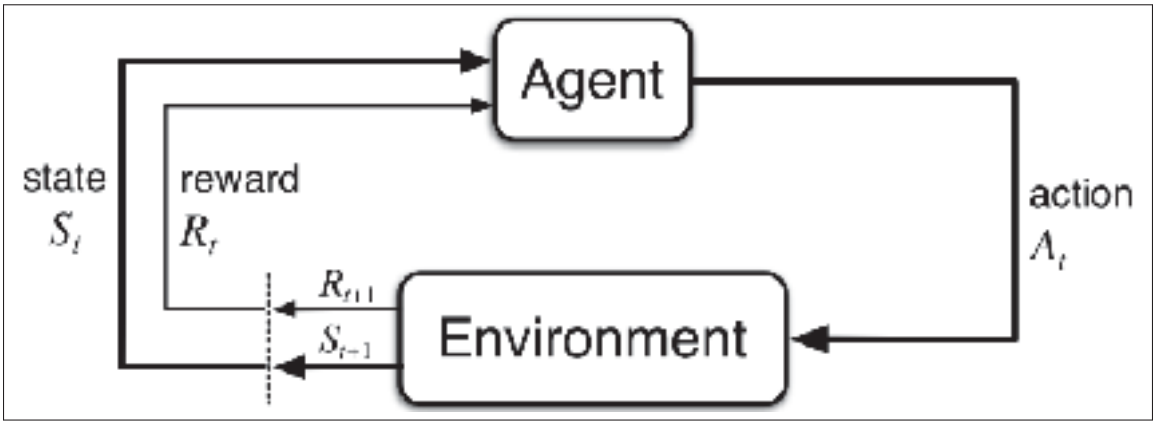


Figure 1.6 Agent-environment interaction in reinforcement learning scenarios
Taken from Sutton & Barto (2018)

can be computed as follows:

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \tag{1.1}$$

There are two primary categories of RL algorithms: Monte Carlo methods, which update the agent's policy only after the episode (for example, a game of chess) is concluded, as well as Temporal Difference (TD) learning, which does not need to wait until the end of the episode and can update the policy after every action. TD learning approaches can be further categorized into on-policy and off-policy algorithms. The most common on-policy algorithm is SARSA (state, action reward, state, action). In SARSA, the agent essentially improves its policy by following, and iteratively updating, the policy. In other words, after each action, the agent updates the value of the state-action pair that it has experienced using the following equation:

$$Q(S_t, A_t) = Q(S_t, A_t) + \alpha \left[R_{t+1} + \gamma\, Q(S_{t+1}, A_{t+1}) - Q(S_t, A_t) \right] \tag{1.2}$$

$Q(S_t, A_t)$ is the reward the agent expects when taking action A_t while in state S_t, and α is a hyperparameter called the learning rate (Sutton & Barto (2018)). When the number of states

and possible actions is reasonably constrained, all the different $Q(S_t, A_t)$ can be represented using a 2-dimensional matrix, called a Q-table.

Q-learning is another popular TD learning approach (Watkins (1989)). Unlike SARSA, Q-learning is an off-policy algorithm, meaning that agents greedily update their policy by selecting the most rewarding future actions without consulting the policy (Sutton & Barto (2018)). The following equation shows how Q-learning updates $Q(S_t, A_t)$ without following the policy:

$$Q(S_t, A_t) = Q(S_t, A_t) + \alpha \left[R_{t+1} + \gamma \max_a Q(S_{t+1}, a) - Q(S_t, A_t) \right] \tag{1.3}$$

Reinforcement Learning is a useful tool for solving the anti-jamming problem due to the fact that many of its implementations are model-free, meaning that agents do not possess a clear representation of how their actions affect the environment, nor do they know in advance the rewards associated with these actions. Instead, purely by trial and error, they can learn this information by themselves and still develop highly performing policies. This concept can be easily applied to anti-jamming since CRs (which can fill the role of agents in RL applied to anti-jamming) do not necessarily know anything about the jammers they are facing or the jamming algorithms that they are using. Nonetheless, if the jammer's behaviour can be predicted, the CRs will discover the patterns behind their behaviour and learn to avoid them.

An example of RL being applied to anti-jamming involves CRs using their sensing data to learn to predict when a channel will be attacked by a jammer, in order to maximize the throughput of the network. Yao & Jia (2019) formulate the anti-jamming problem as a Markov game, and employ a collaborative multi-agent anti-jamming algorithm to detect and avoid a jammer that hops from channel to channel. This algorithm makes use of decentralized Q-learning, meaning that each node in the network maintains and updates its own separate Q-table. The authors demonstrate that their RL scheme performs much better than independent Q-learning, with much faster convergence of the Q-table. However, in order for each agent to choose the optimal action at each time step, it must coordinate with the other nodes in the network and solve a joint optimization problem, which is not feasible in reality.

Similarly, Slimeni *et al.* (2018) also apply Q-learning (though in a modified form) to teach SUs to avoid jammers in a Markov game. However, unlike other RL solutions, they use a reward function that not only penalizes collisions with the jammer, but also unnecessary channel switching. The objective is therefore twofold: avoid the jammer, and transmit on vacant channels for as long as possible before switching. Results show that the members of the network are able to reliably avoid sweeping, reactive and follow-on jammers with performance that exceeds the standard Q-learning algorithm. This solution uses wideband sensing, meaning that the RL agents possess full visibility of the spectrum at all times. While this is possible, the energy needed to simultaneously sense all channels limits the real-life application of this solution. Furthermore, the agent is at first vulnerable to the jammer until the Q-table converges.

Aref & Jayaweera (2017) use two Q-tables, one for selecting channels for sensing and the other for transmission. They not only seek to avoid a sweeping jammer, but also to avoid mutual interference caused by multiple CRs attempting to transmit on the same channel. While the proposed solution is innovative and performs well, evading a single sweeping jammer using RL is a simple problem. Using a more challenging jammer model would have done more to advance the field of RL-based anti-jamming solutions.

Finally, Lundén *et al.* (2013) propose a multi-agent reinforcement learning solution to a scenario where a network of CRs is under attack by a multitude of random jammers. This solution represents the environment as a partially observable Markov decision process, where each member of the network is unable to observe the entire environment at once, forcing them to employ belief states to represent their understanding of the occupancy of the spectrum. Each CR builds a transmission policy based on its observations as well as those of its neighbours in order to determine which channels are safe for transmission. While this solution proposes an interesting way of disseminating sensing information within the network, using an RL-based solution for random jammers limits their performance

1.8 Conclusion

This chapter began by introducing the ongoing problem of spectrum scarcity, as well as the concept of cognitive radios and spectrum sensing as a solution to this problem. Afterwards, we presented the unique military context of this research project, and how it relates to the more general field of spectrum sensing. Then, after going into details regarding jammers, as well as the algorithms they use and the ones used to oppose them, we introduced how reinforcement learning can be used to learn jammers' behaviour and therefore avoid transmitting on jammed channels.

CHAPTER 2

SYSTEM MODEL AND STOCHASTIC GAME FORMULATION

2.1 Introduction

In this chapter, we will be describing the system model used to represent our solution to the joint collaborative spectrum sensing and anti-jamming problem in the context of tactical wireless networks. After introducing some theory, we give a thorough description of the system model, including an overview of the environment we are simulating, as well as the mathematical representation of the jammers. Afterwards, we describe the stochastic game used to represent the interactions of the network's Wireless Nodes (WNs) with the frequency spectrum and with each other.

Following that, we provide an overview the pseudo-random channel selection algorithm we designed to allow WNs to collaboratively scan the spectrum and detect as many jammers as possible. Finally, we describe our solution's super-decision vector scheme where WNs share and combine their sensing information in a way that allows them to better determine which channels are available to use for transmitting data without the risk of being affected by hostile jammers.

2.2 Theoretical Background

In this section, we present the mathematical notions that form the basis of our representation of the multi-agent anti-jamming problem. This includes notions from game theory and probability.

2.2.1 Stochastic Games

A stochastic game is a type of dynamic game that is played in successive stages. The number of stages can be finite or infinite. At each stage, the game is in one of a discrete number of

states. Each player in the stochastic game takes an action at each stage, which yields a reward, or payoff. State transition probabilities, along with the actions taken by each player, will result in the game transitioning from one state to another (Shapley (1953)). A common use for stochastic games is to model real-life phenomena, including economics and political science. If sufficiently robust, these models can simulate complex dynamic interactions and allow their users to make accurate predictions based on their outcome (Solan & Vieille (2015)).

If a stochastic model satisfies the Markov property, i.e. if the probability of reaching a given state given a certain action depends only on the current state (and not any prior state), then this process is called a Markov model (Bickenbach & Bode (2003)). In other words, a Markov model is memoryless. This property makes predictive reasoning and computation much simpler than in non-Markov stochastic processes.

Stochastic games, like many other types of dynamic games, can be zero-sum or non-zero sum. A game is described as zero-sum if the gains (or reward) of one user are exactly balanced by the losses of the other users in the game. Hence, the sum of the players' gains is always equal to 0 (Renault (2019)). Betting games such as Poker are examples of zero-sum games, where the winning player's gains are equal to the amount wagered by the other, losing players.

In a non zero-sum game, the gains of one player do not necessarily result in losses for other players. One such game is the prisoner's dilemma (Bonanno (2018)), summarized in Figure 2.1. For example, two employees at a firm, Doug and Ed, are tied in the running for an employee-of-the-year prize, and the recipient of the prize is the one who puts additional effort into his work before a deadline, sacrificing family time to do so. However, if both put additional effort, or if neither put in additional effort, neither employee will win the prize. If both players act in their own self-interest, neither will win the prize, but both will needlessly sacrifice family time, but if both players reach an agreement to exert normal effort, neither will win the prize but they will not suffer the loss of reduced family time. Non zero-sum games therefore allow a certain level of cooperation between opposing players.

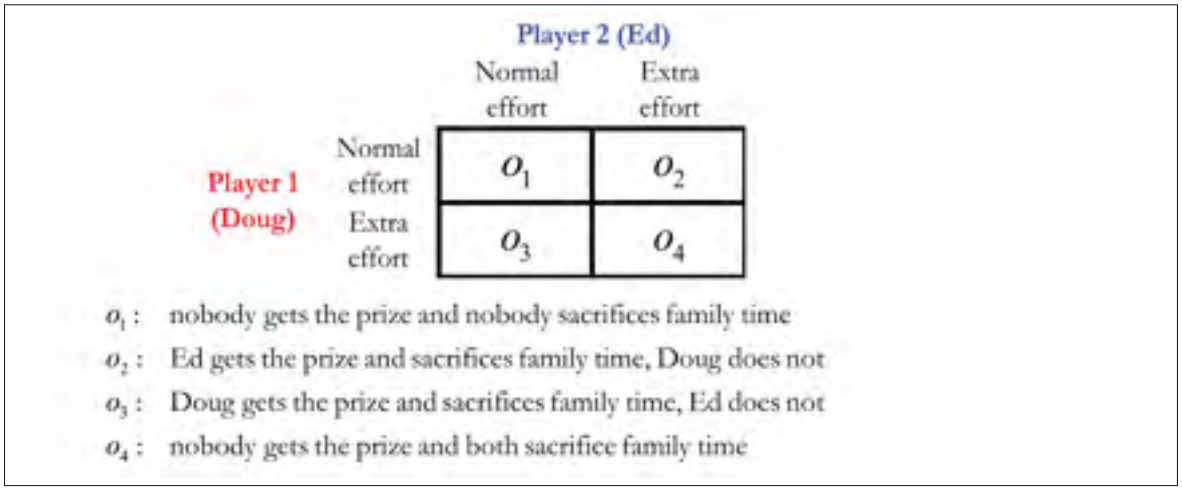


Figure 2.1 Outcomes of prisoner's dilemma
Taken from Bonanno (2018)

2.2.2 Hidden Markov Models

A hidden Markov model (HMM), as its name implies, is a Markov model whose states are not directly observable by the user. The observer can still receive feedback from the environment in the form of observations. One of the major challenges in HMMs is to reconstruct the series of states the environment has been in, based only on the sequence of observations that the user has made. An example of a hidden Markov model is given by Figure 2.2, where B refers to the matrix giving the probability of experiencing each observation in each possible state, X_i is the state at stage i of the HMM, $\mathcal{O}_i$ is the observation made at that moment, and A is the state transition probability matrix.

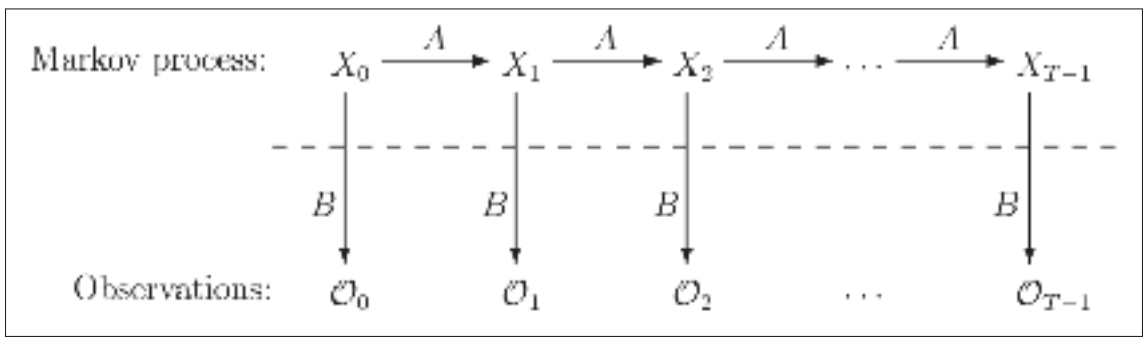


Figure 2.2 Hidden Markov model
Taken from Stamp (2004)

These observations do not necessarily allow the user to definitively reconstruct the series of state transitions, since multiple combinations of states can lead to the same sequence of observations. It is nonetheless possible to associate a probability to each of the different possible sequences of state transitions that led to the observation sequence experienced by the user. It is then possible to identify the most probable state sequence with a degree of certainty equal to the probability associated with that state sequence (Stamp (2004)).

2.3 System Model

We represent an ad hoc network of N_{WN} WNs operating in a deserted, empty terrain such as a field or desert. The spectrum is divided into N_{FB} orthogonal channels, each of equal width. Assigned to each of these channels is exactly one jammer. Each WN possesses a predetermined transmission range, and any WN found within another WN's transmission range is considered to be that WN's neighbour, and vice versa. When two members of the network are neighbours, they can share sensing information with each other.

Our objective is to develop channel selection and data fusion techniques that allow the members of the network to detect as many jammers as possible. A secondary objective is to maximize the number of transmissions that occur on unjammed channels. To test the effectiveness of our solution, we will compare its ability to reach these goals with alternative algorithms used in literature to achieve similar goals.

2.3.1 Assumptions

Below are the assumptions made when developing the system model. We make these assumptions to simplify the implementation of the stochastic game in order to keep the emphasis of the simulation on the joint collaborative spectrum sensing and anti-jamming problem.

- The only entities in the environment that are transmitting on the spectrum are the WNs and the jammers. Furthermore, there is no existing infrastructure in the environment that can fulfill the role of a base station. The WNs are suitable for forming a mesh network.

- The number of channels, WNs, and jammers is constant throughout the simulation.
- All WNs and jammers remain immobile for the duration of the simulation.
- Neighbouring WNs share information with each other using a common control channel that is not subject to fading, collisions, or jamming (Lo & Akyildiz (2012)).
- A WN cannot directly communicate with WNs that are not its neighbours, i.e. those outside the radius given by its transmission power.
- The actions of the WNs are sufficiently synchronized to allow all of their actions to occur simultaneously.
- Each WN always has data to transmit.
- A jammer that is actively attacking a channel has no effect on adjacent channels. Its impact is strictly limited to the channel to which it is assigned, without any spillover.

2.3.2 Probabilities of False Alarm

We represent channel fading using two well-known signal propagation models: additive white Gaussian noise (AWGN) and Rayleigh fading. Both affect the probabilities of experiencing a false alarm, i.e. the probability that a WN incorrectly detects a jammer when the observed channel is in fact vacant. Due to the multipath fading component of the Rayleigh fading model, its impact on the reliability of a WN's observations is considerably higher than when we use the AWGN model.

Neighbouring nodes can mutually support each other by simultaneously sensing the same channel, which increases the diversity order of the sensing action and reduces the probability of false alarm for all the neighbouring nodes sensing that channel. Probabilities of false alarm are represented by $p_{fa,m}$, where m corresponds to the diversity order of the sensing action. In our simulation, we use the same probabilities of false alarm as in Lundén *et al.* (2013).

2.3.3 Probabilities of Detection

The probability of successfully detecting a jammer that is actively attacking a channel is also a function of m. Assuming AWGN channels, this probability $p_{d,m,AWGN}$ is a function of the SNR γ of the signal observed by the WN (Digham *et al.* (2003)). The formula for computing this probability is given by:

$$p_{d,m,AWGN} = Q_{mN/2}\left(\sqrt{\frac{a\gamma}{\sigma^2}}, \sqrt{\frac{\lambda}{\sigma^2}}\right) \tag{2.1}$$

σ^2 is the variance of the observed signal, a is a non centrality parameter, and λ is a decision threshold. $Q_{mN/2}(\cdot,\cdot)$ is the generalized Marcum Q-function of order m (Digham *et al.* (2003)). In the case of Rayleigh fading, we consider that the WN samples the signal N times, which gives an average detection probability of $\bar{p}_{d,Ray}$. Equation (2.2) shows how this probability is computed for a single WN.

$$\bar{p}_{d,Ray} = e^{-\frac{\lambda}{2\sigma^2}} \sum_{i=0}^{N/2-1} \frac{\left(\frac{\lambda}{2\sigma^2}\right)^i}{i!} + \left(\frac{2\sigma^2 + a\bar{\gamma}}{a\bar{\gamma}}\right)^{N/2-1} \times \left(e^{-\frac{\lambda}{2\sigma^2+a\bar{\gamma}}} - e^{-\frac{\lambda}{2\sigma^2}} \sum_{i=0}^{N/2-1} \frac{\left(\frac{\lambda a\bar{\gamma}}{2\sigma^2(2\sigma^2+a\bar{\gamma})}\right)^i}{i!} \right) \tag{2.2}$$

When the diversity order is greater than 1, we obtain the improved detection probability by combining the values of $\bar{p}_{d,Ray}$ of each collaborating member of the network:

$$\bar{p}_{d,m,Ray} = 1 - \prod_{i=1}^{m} (1 - \bar{p}_{d,Ray,i}) \tag{2.3}$$

2.3.4 Two-State Markov Process Representation of the Jammer

We represent the jammers using a two-state Markov model. At any given moment, a jammer may be idle, or it may be actively attacking the channel. At each time step, it may remain in its current state, or it may switch to the alternate state. We define $p_{k,00}$ as the probability that a jammer k stays in the idle state from one time step to the next. Conversely, we define $p_{k,11}$ as the probability that it stays in the active state. From these probabilities, it follows that $p_{k,01} = 1 - p_{k,00}$ represents the probability that the jammer k switches from idle to active, and that $p_{k,10} = 1 - p_{k,11}$ represents the opposite. Figure 2.3 summarizes the jammer Markov model.

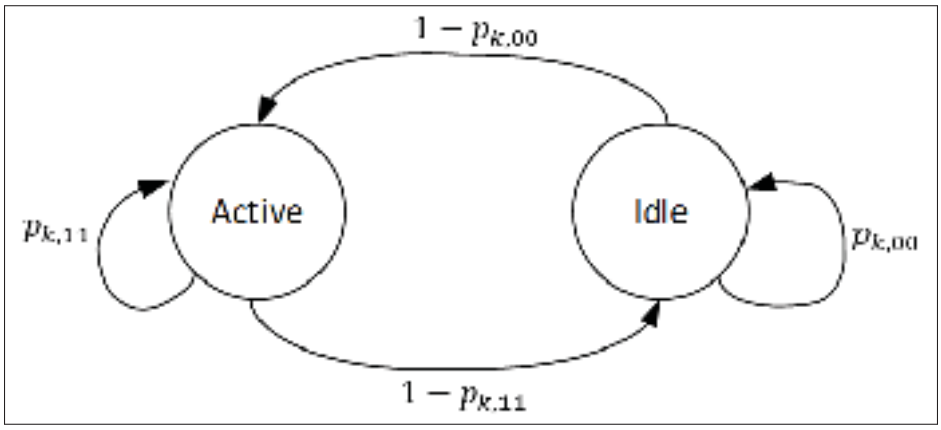


Figure 2.3 Representation of jammer Markov model

2.3.5 Network Configuration

The configuration of the network will affect the number of neighbours that each WN possesses. Additionally, this will affect each WN's distance to each jammer: if a WN is far away from the jammers, the power that it detects on the channel will be lower than if it was closer to the jammers, resulting in a lower observed SNR γ and thus a lower detection probability.

The network configuration used in our simulations is given in Figure 2.4. Each black dot represents a WN. Dotted lines between two WNs indicate that the pair of WNs are neighbours.

All of the jammers are grouped together in the red box, and the limit of their transmission range is given by the red circle.

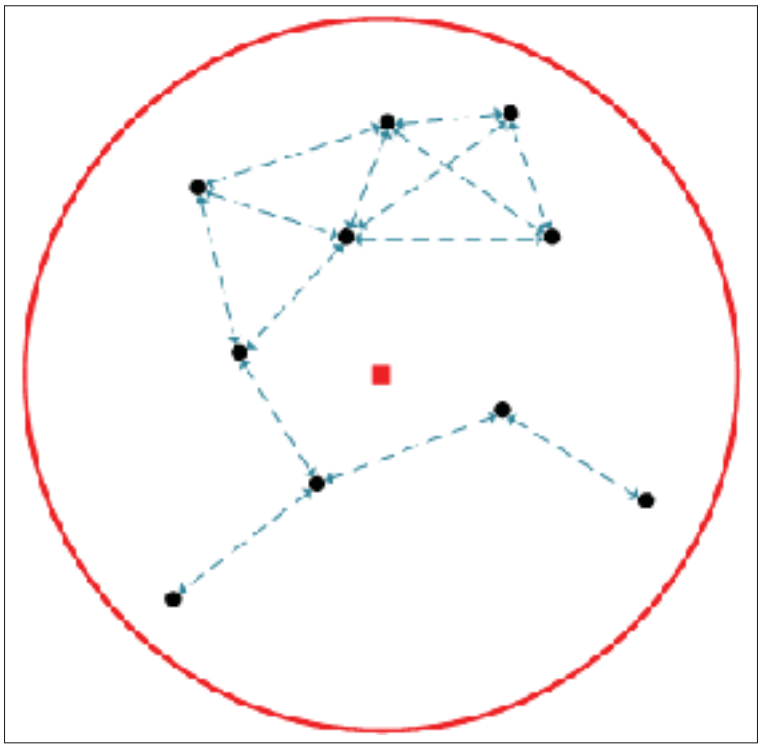


Figure 2.4 Configuration of the tactical wireless network used in the simulation

2.4 Stochastic Game Model

We represent the anti-jamming problem as a partially observable stochastic game modeled as follows:

- A series of time steps $t = 1, 2, \ldots, T$ that are each of fixed length and split into three sub-slots: sensing, collaboration, and transmission.
- A set of Wireless Nodes $M = \{1, \ldots, N_{WN}\}$ dispersed throughout the environment.
- A set of N_{FB} channels in the frequency band. Each WN senses exactly one of these channels at each time step t and observes its occupancy.

- A set of possible occupancy values for a channel j sensed by WN i at time step t, given by $s_t^{i,j} \in \{vacant, occupied\}$.

- The vector $\boldsymbol{s}_t^i$ represents the occupancy of each channel sensed by WN i at time step t, given by $\boldsymbol{s}_t^i = [s_t^{i,1}, \ldots, s_t^{i,N_{FB}}]$. Its value is a member of the set $\{vacant, occupied\}$.

- The state $\boldsymbol{s}_t$ is an array of vectors $\boldsymbol{s}_t^i$, showing each WN's perception of each channel at time t, given by $\boldsymbol{s}_t = [\boldsymbol{s}_t^1, \ldots, \boldsymbol{s}_t^{N_{WN}}]$.

- A set of possible sensing actions $a_t^i \in A_i$, where $i \in M$, and $A_i = \{1, \ldots, N_{FB}\}$ represents which channel will be sensed by the WN at time step t.

- τ_t^i represents an observation made by WN i at time t, i.e. it is the observation made from taking action a_t^i. Its value is a member of the set $\{vacant, occupied\}$. Due to the imperfect nature of the WNs' observations, this value may not reflect the actual occupancy of the channel as given by $s_t^{i,j}$.

- A vector of sensing decisions made by each WN i $\boldsymbol{d}_t^i \in \boldsymbol{D}_{i,t}$. Each element of $\boldsymbol{d}_t^i$ is a member of the set $\{vacant, occupied\}$.

- A vector of super-decision vectors $\boldsymbol{\delta}_t^i$ obtained by combining WN i's decision vector $\boldsymbol{d}_t^i$ with its neighbours' decision vectors $\boldsymbol{d}_t^j \; \forall \; j$ into a single vector. Like $\boldsymbol{d}_t^i$, each element of $\boldsymbol{\delta}_t^i$ is a member of the set $\{vacant, occupied\}$.

- A transmission outcome $x_t^i \in \{successful, jammed\}$ representing the outcome of the transmission performed by WN i at time step t.

- A state transition function ϕ: $\boldsymbol{S} \times \boldsymbol{A} \times \boldsymbol{S} \rightarrow \mathbb{R}$ that defines the state transition probabilities $P(\boldsymbol{s}_{t+1} \mid \boldsymbol{s}_t, a_t^1, \ldots, a_t^{N_{WN}})$. We use the jammer probabilities $p_{k,00}$ and $p_{k,11}$ as the state transition probabilities. We assume that the agents' actions do not affect the state transition probabilities: $P(\boldsymbol{s}_{t+1} \mid \boldsymbol{s}_t, a_t^1, \ldots, a_t^{N_{WN}}) = P(\boldsymbol{s}_{t+1} \mid \boldsymbol{s}_t)$. In other words, we assume that a WN's sensing action does not cause any interference on the channel that could negatively affect the observation of another WN.

2.4.1 Sensing, Cooperation, and Fusion Algorithm

The sensing, cooperation, and fusion algorithm makes up the heart of the simulation software. Each iteration of the algorithm's *while* loop corresponds to one time step which, as mentioned previously, is split into three sub-slots: sensing, collaboration, and transmission, corresponding to the three operations conducted by each WN in each of the T time slots of the simulation.

Upon initialization, each jammer is randomly assigned to one of the two possible states: active or idle, and each WN is randomly assigned an initial action a_0^i, corresponding to the channel that it will initially sense. The jammers are assigned predetermined randomly generated numbers between bounds of 0.85 and 0.98 as their state transition probabilities $p_{k,00}$ and $p_{k,11}$. Then, at each time step of the simulation, each WN i performs energy detection on exactly one of the N_{FB} channels in the spectrum to determine if a jammer is actively attacking that channel. This sensing action yields an observation datum τ_t^i, which the WN i shares with each of its neighbouring WNs j in return for their own sensing observations. Each WN i also tells its neighbours on which channel the observation took place. Each wireless node i then applies a data fusion rule on these observations and combines them into a decision vector $\boldsymbol{d}_t^i$, indicating whether it believes each channel is occupied by a jammer or if it is safe for transmission. Then, based on its observation in the previous time step, the WN will choose its next action a_{t+1}^i, i.e. which channel it will sense in time step $t+1$.

Next, in order to obtain a clearer picture of the utilization of the spectrum by the jammers, each WN i performs a second round of data collaboration by sharing its decision vector $\boldsymbol{d}_t^i$ with each of its neighbours j. Each WN applies the data fusion rule for a second time, and combines the decision vectors into a super-decision vector $\boldsymbol{\delta}_t^i$. Finally, each WN consults its super-decision vector and selects a channel that it believes to be vacant and uses it to broadcast data. This is summarized in Algorithm 2.1. The order in which the WNs iterate through the two *for* loops is arbitrary and may be randomized at each time step.

It is important to note that a WN will not attempt to transmit data if it believes that all N_{WN} channels are jammed. Additionally, it will ignore channels on which it has no information

pertaining to their occupancy. Instead, the WN will skip the transmission phase and wait until the beginning of the next time step.

Algorithm 2.1 Sensing, cooperation, and fusion algorithm

Initialize $t = 0$
Initialize set of Wireless Nodes M
Initialize set of Jammers J
Initialize set of Channels W
Initialize $a_0^i = random(1,\ldots,N_{FB}),\ i \in \{1,\ldots,N_{WN}\}$
Initialize $w_0^z = bernoulli(0.5),\ z \in \{1,\ldots,N_{FB}\}$
$\boldsymbol{s}_0 = computeOccupancy(M,J,W)$
while $t < T$ **do**
    // Sensing phase
    $\tau_t^i = sense(a_t^i),\ i \in \{1,\ldots,N_{WN}\}$
    // Collaboration phase
    for *each WN i in M* **do**
        Transmit tuple $\{a_t^i, \tau_t^i\}$ to neighbours
        for *each neighbour j of WN i* **do**
            j receives $\{a_t^i, \tau_t^i\}$ from i
            j adds a_{t+1}^i to its action vector: $\boldsymbol{a}_{t+1}^j = \boldsymbol{a}_{t+1}^j \cup a_{t+1}^i$
        end
    end
    $\boldsymbol{d}_t^i = fusion(\tau_t^i, \{\tau_t^j\}_{\forall j}), i \in \{1,\ldots,N_{WN}\}$
    $a_{t+1}^i = chooseAction(\tau_t^i, \boldsymbol{a}_t^i), i \in \{1,\ldots,N_{WN}\}$
    for *each WN i in M* **do**
        Transmit decision vector $\boldsymbol{d}_t^i$ to neighbours
        for *each neighbour j of WN i* **do**
            j receives $\boldsymbol{d}_t^i$ from i
        end
        $\boldsymbol{\delta}_t^i = fusion(\boldsymbol{d}_t^i, \{\boldsymbol{d}_t^j\}_{\forall j})$
        // Transmission Phase
        $x_t^i = transmit(\boldsymbol{\delta}_t^i)$
    end
    $\boldsymbol{s}_{t+1} = computeOccupancy(M,J,W)$
    $t = t + 1$
end

2.4.2 Collaborative Pseudo-Random Channel Selection

One of the two principal contributions of this research project is the algorithm employed by the WNs to choose which channel to sense. We call this the collaborative pseudo-random channel selection algorithm, on account of the fact that it exploits the random nature of the jammers' behaviour, as well as the fact that it encourages collaboration between WNs to increase sensing reliability.

To develop this solution, we repurposed the distributed spectrum sensing information time slot scheduling algorithm from Lundén *et al.* (2015). The original use for this algorithm was to develop a schedule for WNs to share sensing observations with each other without causing collisions on the control channel (for simplicity's sake, we assume no such collisions are possible in our system model). We discovered that this algorithm could be applied to collaborative spectrum sensing, with surprising results.

Algorithm 2.2 Collaborative pseudo-random channel selection algorithm

```
1  if τ_t^i == occupied then
2  |   a_{t+1}^i = a_t^i
3  end
4  else
5  |   if u(0,1) ≤ ε_n then
6  |   |   a_{t+1}^i = a_t^j
7  |   end
8  |   else
9  |   |   a_{t+1}^i = random(A_i \ {a_t^i, a_t^j ∀ j})
10 |   end
11 end
```

The collaborative pseudo-random channel selection algorithm is given in Algorithm 2.2. To summarize, if a WN detects a jammer on a channel at time step t (or rather, if its observation $\tau_t^i == occupied$, regardless of whether the jammer is in fact active or if the WN's observation is a false alarm), then it will observe that channel again at time step $t+1$. Due to the reasonably high values of $p_{k,00}$ and $p_{k,11}$, jammers are more likely to remain in their current state than to

transition to the alternate state. As a result, if a WN detects a jammer (assuming the observation is not a false alarm), then there is a high probability that the jammer will continue attacking the channel in the next time step. Seeing how the WNs' goal is to detected as many jammers as possible, this justifies continuing to sense that channel for as long as a jammer is detected on it.

If the WN does not observe that the channel is under attack by a jammer, then we enter an exploration-exploitation trade-off situation, controlled by a hyper-parameter ε_n. This constant is equivalent to a probability that the WN chooses the same action as one of its neighbours, in order to increase the diversity order and therefore the sensing reliability for both members of the network.

This exploitation scenario has two potential benefits: if the neighbour j is repeatedly observing false alarms, then a higher diversity order would reduce the false alarm probability $p_{fa,m}$ and finally allow j to see that the channel is in fact vacant, causing it to cease observing this channel. The second benefit is in case the jammer is active: a higher diversity order reduces the probability of a missed detection, thus allowing j to stay on an active jammer for longer, instead of erroneously believing that the channel is vacant, which would force it to stop sensing that channel and start sensing elsewhere. Furthermore, if a WN mistakenly observes a channel as vacant (and this observation is not corrected during the collaboration phase), then it may transmit on that channel, which is still under attack by a jammer, resulting in an intercepted transmission.

Consequently, there is a probability $1 - \varepsilon_n$ that the WN will select a channel that is not being sensed by itself or by any of its neighbours. In other words, it attempts to expand the subset of channels being sensed by this group of WNs. By exploring a larger range of channels, WNs can potentially detect more active jammers, or find vacant channels that can be used for transmission. The value of ε_n is therefore a trade-off between sensing reliability and the number of channels observed by a node and its neighbours.

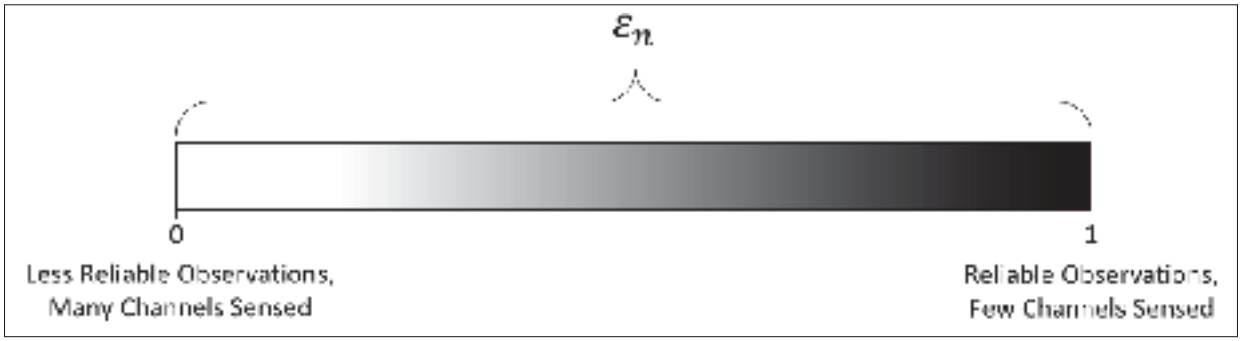


Figure 2.5 Representation of the exploration-exploitation trade-off

2.4.3 Super-Decision Vectors

The second contribution of this research project consists of the network's collaboration and data fusion scheme. As described earlier, each WN i shares its sensing information τ_t^i with each of its neighbours j, it applies a data fusion rule and merges its set of observations into a decision vector $\boldsymbol{d}_t^i$ that the WN can consult to find a vacant channel for transmission. We expand upon this concept by adding a second stage to the collaboration phase: after each WN derives its decision vector, it shares this vector with each of its neighbours, exactly like it did earlier with its sensing information. Each WN then applies the data fusion rule once again, resulting in a super-decision vector $\boldsymbol{\delta}_t^i$.

Super-decision vectors allow a WN to indirectly access sensing information not only from its neighbours, but from its neighbours' neighbours as well. If a WN i has a neighbour j, and j has a neighbour k that is too far from i to be considered its neighbour, then j will incorporate k's observation τ_t^k into its decision vector $\boldsymbol{d}_t^j$. However, when the node begins the second stage of the collaboration phase, j shares its decision vector with its neighbours, including WN i. This means that when WN i synthesizes its super-decision vector $\boldsymbol{\delta}_t^i$, it not only uses information from itself and its neighbour j, but also from k. In other words, using super-decision vectors increases the range of shared information from one hop to two. Figure 2.6 summarizes this phenomenon. Another useful advantage of this solution lies in the fact that WNs that are further away from the jammers (which means that they perceive the jammers' signal with a

lower SNR, and thus a lower probability of detection) can receive sensing information from neighbours who are closer to the jammers and thus observe the occupancy of the spectrum with a higher degree of certainty.

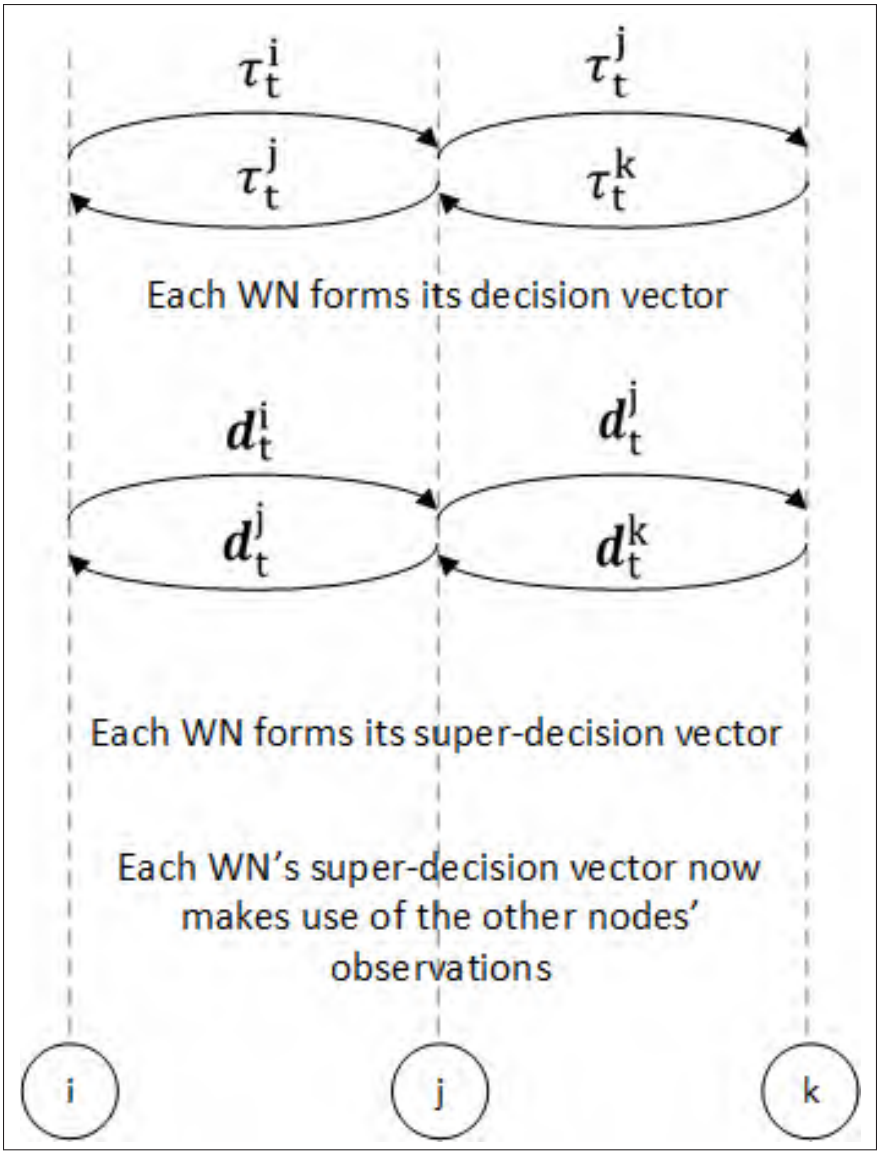


Figure 2.6 Representation of two-hop data sharing

The WNs use the OR-rule when performing data fusion. This means that only a single positive observation (i.e. an observation that detected a jammer) is needed for a WN to determine that a channel is jammed. In other words, the observation does not need to be corroborated by other

nodes' observations as would be the case if we were using the Majority-rule or the AND-rule. While this fusion rule is the best in terms of rapidly propagating observations throughout the network, it suffers from an important drawback: false alarms can contaminate super-decision vectors that are as far as two hops away. For example, if two neighbouring nodes i and j both simultaneously sensed the same vacant channel, and i fell victim to a false alarm, it would share this incorrect sensing data with j, even if j correctly sensed the channel as vacant. Not only would this contaminate j's decision vector, but also its super-decision vector as well as those of its neighbours. This reduces the number of available channels that the WN believes to be safe for transmission, thereby limiting its ability to transmit data. Although a higher diversity order m reduces the probability of a false alarm, the risk cannot be erased completely.

2.5 Conclusion

In this chapter, we introduced the basic theory behind stochastic games and Markov processes as well as the system model used to represent the multi-agent anti-jamming problem and the stochastic game that controls the actions of the wireless nodes. We then introduced the primary elements of our solution, i.e. the channel selection algorithm that allows WNs to cooperate with their peers to ensure reliable sensing, as well as the data sharing and collaboration schemes that gives rise to super-decision vectors. Together, these two elements form the basis of our research project's original contributions.

CHAPTER 3

SIMULATION RESULTS

3.1 Introduction

In this chapter, we evaluate the performance of our collaborative channel selection algorithm as well as the inter-node data collaboration and fusion scheme that gives rise to super-decision vectors. We implement the stochastic game described in Chapter 2 and use it to simulate an ad hoc network that is under attack by hostile jammers. We developed two main performance metrics that are used to quantify the effectiveness of our solution. These are described in the following section. We run the simulation for a set number of time steps, and then re-run the simulation a number of times before computing the average performance of each time step in order to reduce the variance of the results.

In all cases, we compare the effectiveness of our solution against random actions (meaning that channel selection for both sensing and transmission are done randomly) as well as the multi-agent reinforcement learning (MARL) scheme described in Lundén *et al.* (2013). In this article, each WN attempts to devise its own transmission policy by using its own sensing observations as well as those of its neighbours to approximate the probabilities that each channel is jammed at any given moment. Much like in our research project, the environment is only partially observable, forcing each WN to employ belief states to represent its understanding of the occupancy of the spectrum.

3.2 Performance Metrics

We consider two performance metrics. First, in order to gauge our solution's ability to detect active jammers, we compute the jammer detection ratio (JDR), which represents the network's effectiveness at finding active jammers. Equation (3.1) represents how we calculate the JDR.

$$JDR = \frac{\text{Number of times a jammer was detected}}{\text{Total number of jams up to time step } t} \tag{3.1}$$

It is worth noting that if a jammer is detected more than once during the same time step, it will only be counted once. In other words, the JDR is the fraction of all instances of jamming that were detected by at least one member of the ad hoc network.

The second performance metric considered in this research project is the WNs' ability to make use of their sensing data, as well as those of their neighbours, in order to identity vacant channels that can be used for transmission. We call this the transmission success rate (TSR). To this end, we tally the total number of successful transmissions (i.e. transmissions that occurred on vacant channels) and divide it by the number of times that the WNs have entered the transmission phase up to time step t, as summarized in Equation (3.2).

$$TSR = \frac{\text{Number of successful transmissions}}{N_{WN} \times t} \tag{3.2}$$

As mentioned previously, WNs will not attempt to transmit if they have not identified at least one vacant channel. Whenever this occurs, the WN will delay its transmission until its super-decision vector contains at least one vacant channel. If the transmission is successful, then the number of successful transmissions is increased by 1 plus the number of deferred transmissions. In other words, the WN transmits the entire contents of its buffer as soon as it is able.

3.3 Simulation Environment

The simulation consists of a Python script written from scratch. The script begins by instantiating all of the N_{WN} Wireless Node objects, and assigning a set of hard-coded coordinates to each of them in order to obtain the network configuration seen in Figure 2.4. The program then instantiates each of the N_{FB} Jammer objects, with their respective probabilities $p_{k,00}$ and

$p_{k,11}$. The script then enters the main loop described in Algorithm 2.1. While it is iterating through the loop, the script computes the JDR or the TSR for each time step t. The simulation is restarted for a predetermined number of epochs. After the final epoch is concluded, we compute the average TSR or JDR of each time step t. This final array of values is then outputted to the user.

Transmitted signals attenuate at a rate of $P_T \times (d/0.05)^{-2.3}$. We choose the exponent -2.3 to represent a flat, deserted environment, we also use a reference distance of $0.05km$. The parameter d represents the Euclidean distance between the transmitter and the receiver. We used the following false alarm probabilities: $p_{fa,1} = 0.0015$, and $p_{fa,m} = 10^{-7}$ for $m \geq 2$ over AWGN channels, as well as $p_{fa,1} = 0.83$, $p_{fa,2} = 0.32$, $p_{fa,3} = 0.03$, $p_{fa,4} = 0.003$, and $p_{fa,m} = 0.001$ for $m \geq 5$ over Rayleigh fading channels (Lundén *et al.* (2013)).

In all simulation scenarios, the jammers' transmission power is set to 15 dB. For received SNR γ ranging from 0 dB to 15 dB, and diversity orders m ranging from 1 to 6 WNs simultaneously sensing the same channel, we use Equation (2.1) to calculate a two-dimensional matrix that the simulation software can consult in order to determine the correct detection probability $p_{d,m,AWGN}$ whenever a WN senses a channel.

Moreover, using the SNR range, we used Equation (2.2) to obtain a one-dimensional array giving the Rayleigh probabilities of detection $\bar{p}_{d,Ray}$ when $m = 1$. Detection probabilities with $m > 1$ are calculated dynamically during the simulation using Equation (2.3). In all cases, the variance of the signal σ^2 is equal to 1, the non centrality parameter a is set to 2, and the decision threshold λ is set to 12.1 in order to approximate probabilities seen in Lundén *et al.* (2013). Remaining simulation parameters are provided in Table 3.1.

3.4 Results

We compare the jammer detection ratio and transmission success rate of three different channel selection techniques: our collaborative pseudo-random channel selection, multi-agent re-

Table 3.1 Table of simulation parameters

Parameter	Values
Number of Wireless Nodes N_{WN}	10
Number of Channels and Jammers N_{FB}	10 or 20
Number of Time Steps T	2000
Jammer Transmission Power	15 dB
Wireless Node Transmission Power	7 dB
Noise Power	1 dB
Exploration-Exploitation Trade-off Constant ε_n	0.1

inforcement learning (Lundén *et al.* (2013)), and random channel selection. We consider two scenarios, one with 10 channels and 10 WNs, and another with 20 channels and 10 WNs.

3.4.1 Jammer Detection Ratio

In addition, we evaluate the impact of sensing reliability on each channel selection technique by running each simulation over AWGN channels, and again over Rayleigh fading. Results are displayed in Figures 3.1 and 3.2, which respectively represent the two scenarios mentioned previously.

We notice in Figure 3.1a that the collaborative pseudo-random channel selection algorithm outperforms the MARL scheme as well as random action selection. The algorithm takes advantage of the fact that, due to the high values of the their state transition probabilities $p_{k,00}$ and $p_{k,11}$, jammers are likely to remain in their current state from one time step to the next, instead of transitioning to the alternate state. WNs that successfully detect a jammed channel are therefore likely to continue detecting that same jammer for as long as the jammer remains in the active state. When the channel finally becomes vacant, the WN will cease sensing that channel and sense the same channel as one of its neighbours (depending on the probability ε_n) or explore the spectrum and sense a channel that is not being observed by the WN or by its neighbours.

Due to the unpredictable nature of the jammers' Markov model, it is impossible to predict when a jammer will switch states. As mentioned in Chapter 1, several other research projects

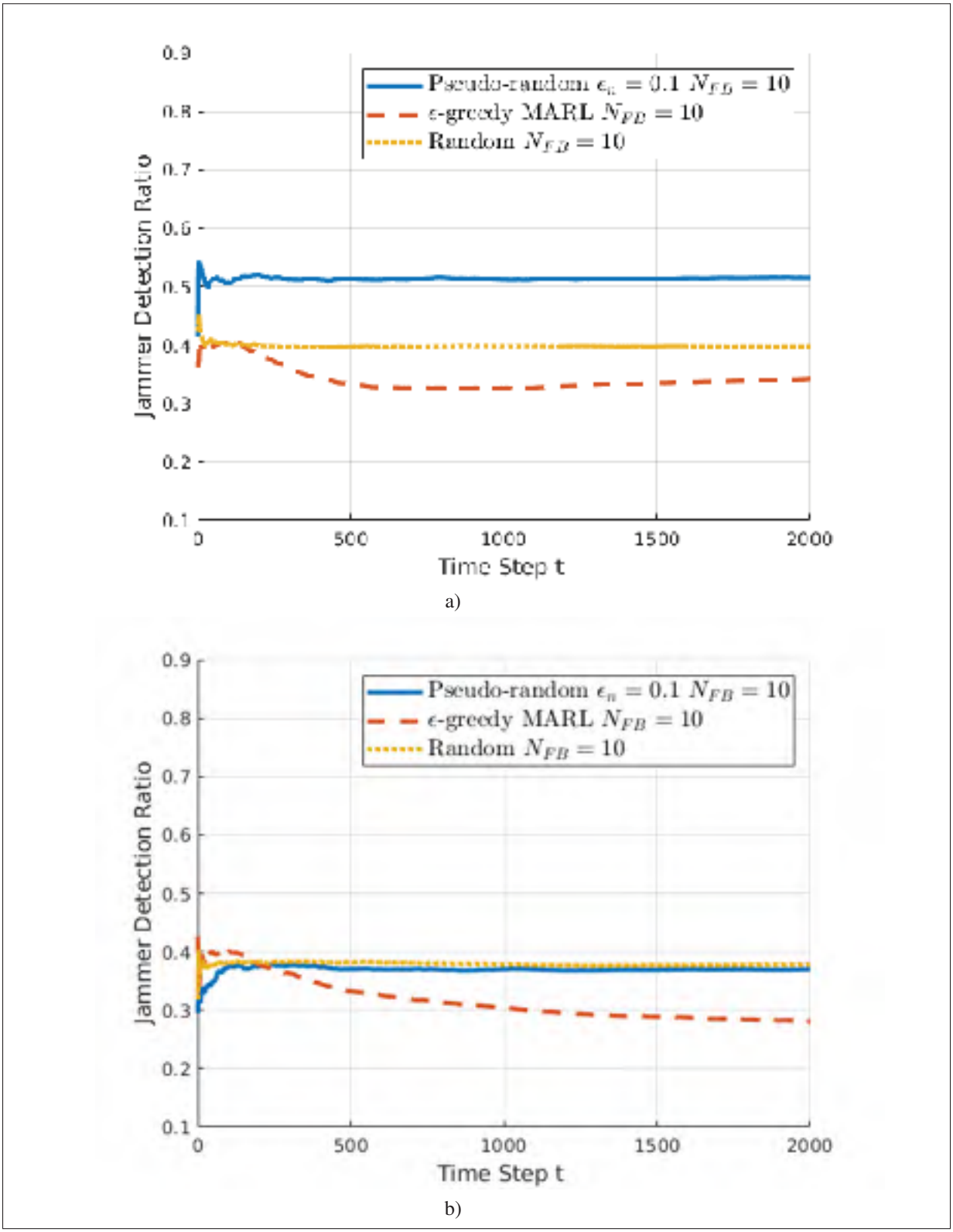


Figure 3.1 Performance evaluation using jammer detection ratio with 10 WNs and 10 channels using (a) AWGN channels and (b) Rayleigh fading

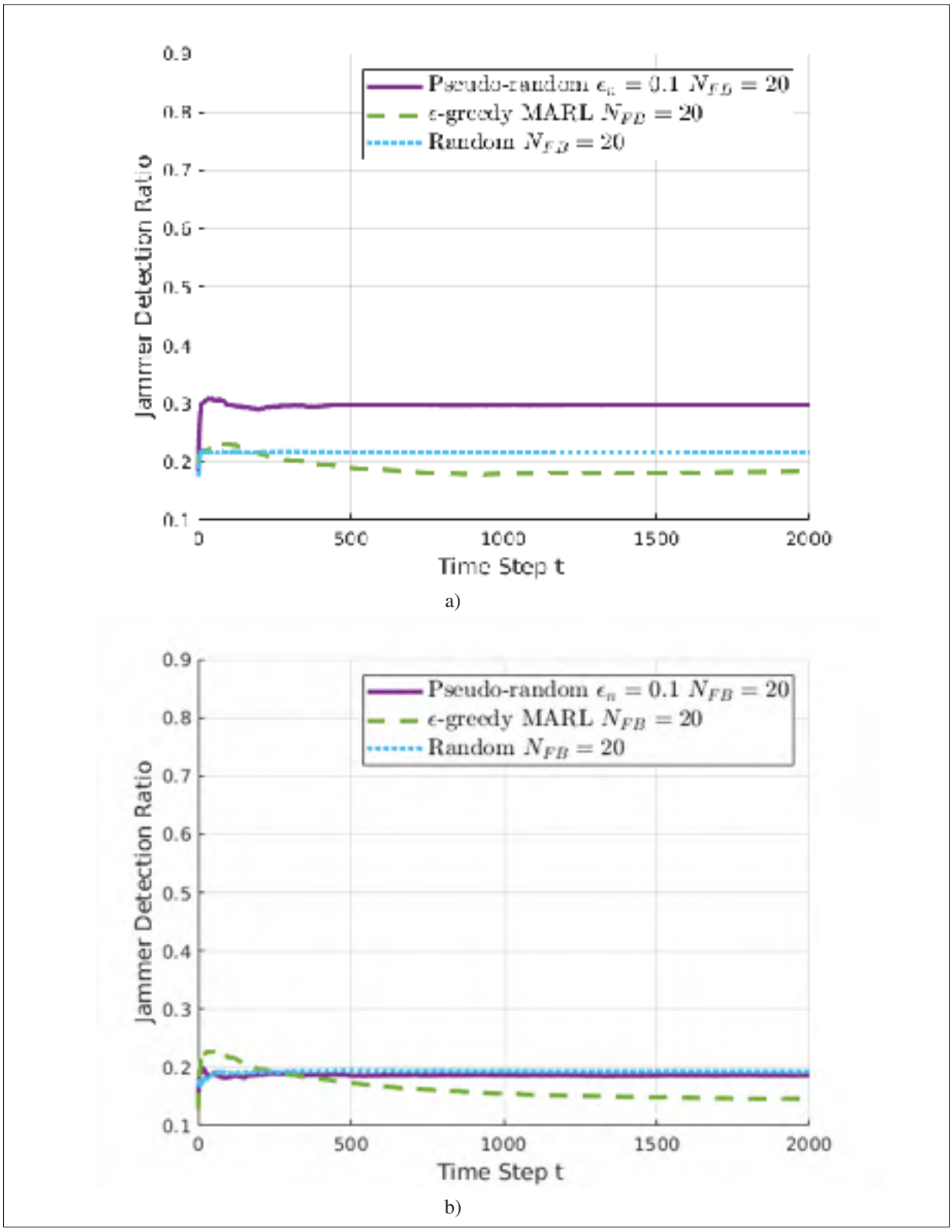


Figure 3.2 Performance evaluation using jammer detection ratio with 10 WNs and 20 channels using (a) AWGN channels and (b) Rayleigh fading

such as Slimeni *et al.* (2018) and Aref & Jayaweera (2017) model the jammer's behaviour as being predictable, which allows RL schemes to learn their behaviour and reliably predict their actions. The random nature of our jammer model may explain why random action selection outperforms the MARL scheme.

In Figure 3.1b, we see that the collaborative pseudo-random channel selection algorithm performs much worse when the channels undergo Rayleigh fading instead of AWGN, to the point where its performance is slightly below random channel selection. Sensing observations are much less reliable in Rayleigh fading, which can cause the WN to observe a jammed channel as being vacant, which causes it to cease sensing that channel and switch to a different channel as per Algorithm 2.1. This can result in the WN switching from sensing a jammed channel to sensing one that is vacant. If the WN successfully detects this channel as being vacant, it must again switch channels. On the other hand, if the WN incorrectly detects the channel as being jammed, it will continue to sense the channel until it makes a correct observation. In either case, a higher diversity order m will improve the sensing reliability and prevent WNs from entering these situations that hinder their performance. In other words, it is necessary to consider the environment's sensing reliability when setting the value of the the exploration-exploitation trade-off constant ε_n.

In Figure 3.2, we observe similar results as when $N_{FB} = 10$ in the sense that the relative performance of each algorithm is essentially unchanged, except for the fact that each curve is roughly 50% lower than its counterpart in Figure 3.1. This can be explained by the fact that when we double the number of channels from 10 to 20, we double the denominator of Equation (3.1), meaning that the number of potential jams is also doubled and the jammer detection ratio is reduced by one half.

3.4.2 Transmission Success Rate

To illustrate our solution's ability to identify and transmit on vacant channels, we compare the TSR when WNs use super-decision vectors $\boldsymbol{\delta}_t^i$ to choose their transmission channel versus

only using local decision vectors $\boldsymbol{d}_t^i$. As we did with the JDR, we compare the TSR of each of the three channel selection algorithms when $N_{FB} = 10$ as well as when $N_{FB} = 20$. It is worth noting that when discussing the TSR, random channel selection means that the WN selects its *transmission* channel randomly, while in the context of the JDR, it refers to randomly choosing the *sensing* channel. If the WN chooses its transmission channel randomly, then it does not make use of its sensing observation at all. Finally, we only use AWGN channels to represent the spectrum when determining the TSR. Results are given by Figure 3.3 for the scenario where $N_{WN} = 10$ and $N_{FB} = 10$ and by Figure 3.4 for the scenario where where $N_{WN} = 10$ and $N_{FB} = 20$.

We observe in Figure 3.3a that the collaborative pseudo-random channel selection algorithm performs better than the MARL scheme and much better than random channel selection. The same can be observed in Figure 3.3b. Comparing our solution's plots in both figures, we see that using super-decision vectors $\boldsymbol{\delta}_t^i$ to select transmission channels yields a slightly higher TSR than using local decision vectors $\boldsymbol{d}_t^i$.

In Figure 3.4, we observe that the relative performance of each algorithm remains the same when using 20 channels versus using 10 channels. Additionally, the performance of each algorithm does not drastically drop when increasing N_{FB} from 10 to 20, unlike with the JDR. This can be explained by the fact that each WN only transmits on channels that it believes to be vacant. Since the configuration of the network stays the same in both scenarios, the number of sensing data that each node possesses does not vary with respect to the number of channels in the spectrum. However, increasing N_{FB} means that each WN now has a greater range of channels that it can sense, which can reduce the average diversity order m of each sensing action, resulting in less reliable observations across the network. This may explain why the TSR of each algorithm is slightly lower when using 20 channels versus using 10.

In both figures, we observe that the performance of each algorithm does not exceed a threshold of roughly 70%. Nonetheless, the TSR obtained using $\boldsymbol{\delta}_t^i$ is slightly higher than when $\boldsymbol{d}_t^i$ is used. In this context, it is important to note that the simulation does not take into account the

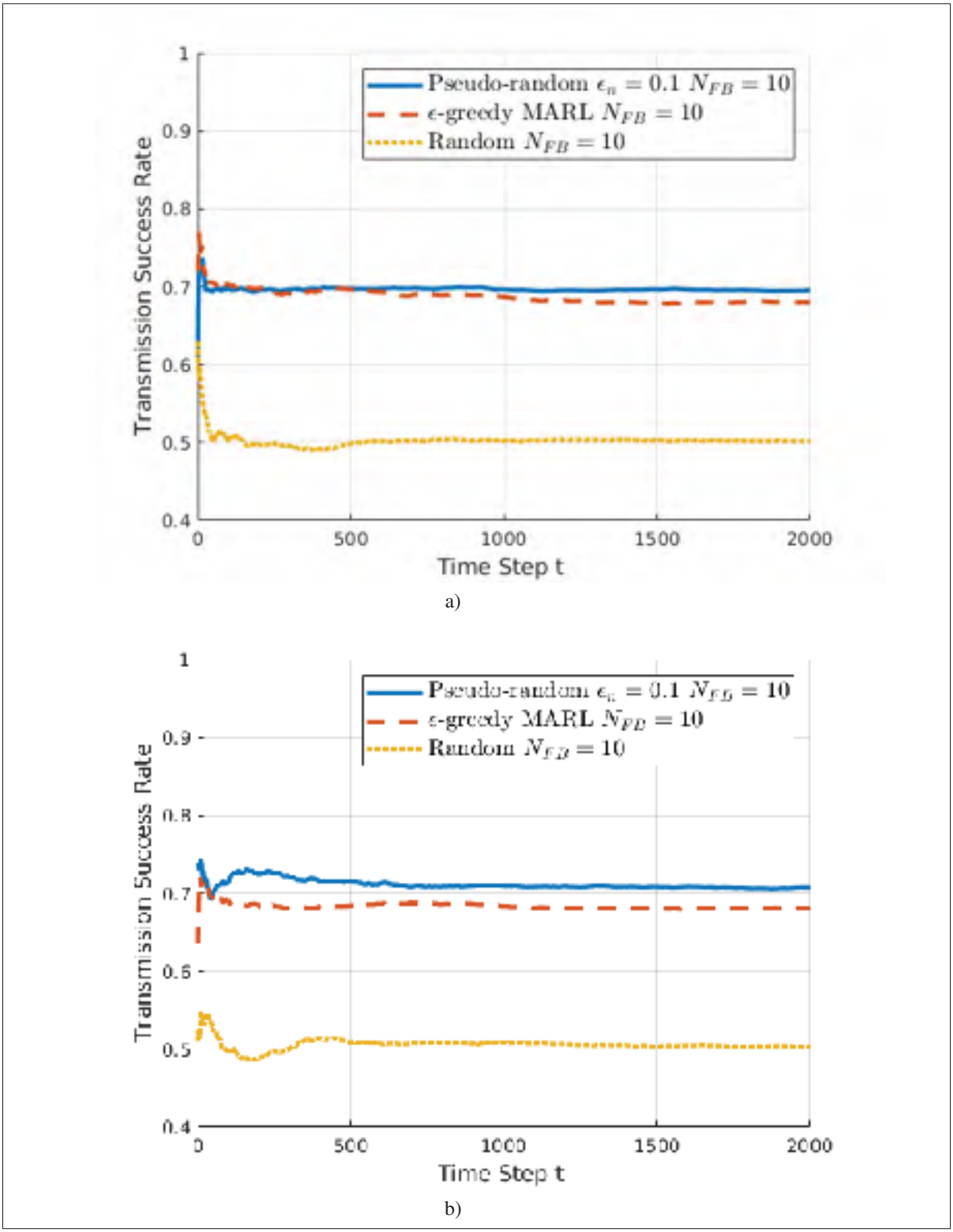


Figure 3.3 Performance evaluation using transmission success rate with 10 WNs and 10 channels using (a) local decision vectors and (b) super-decision vectors

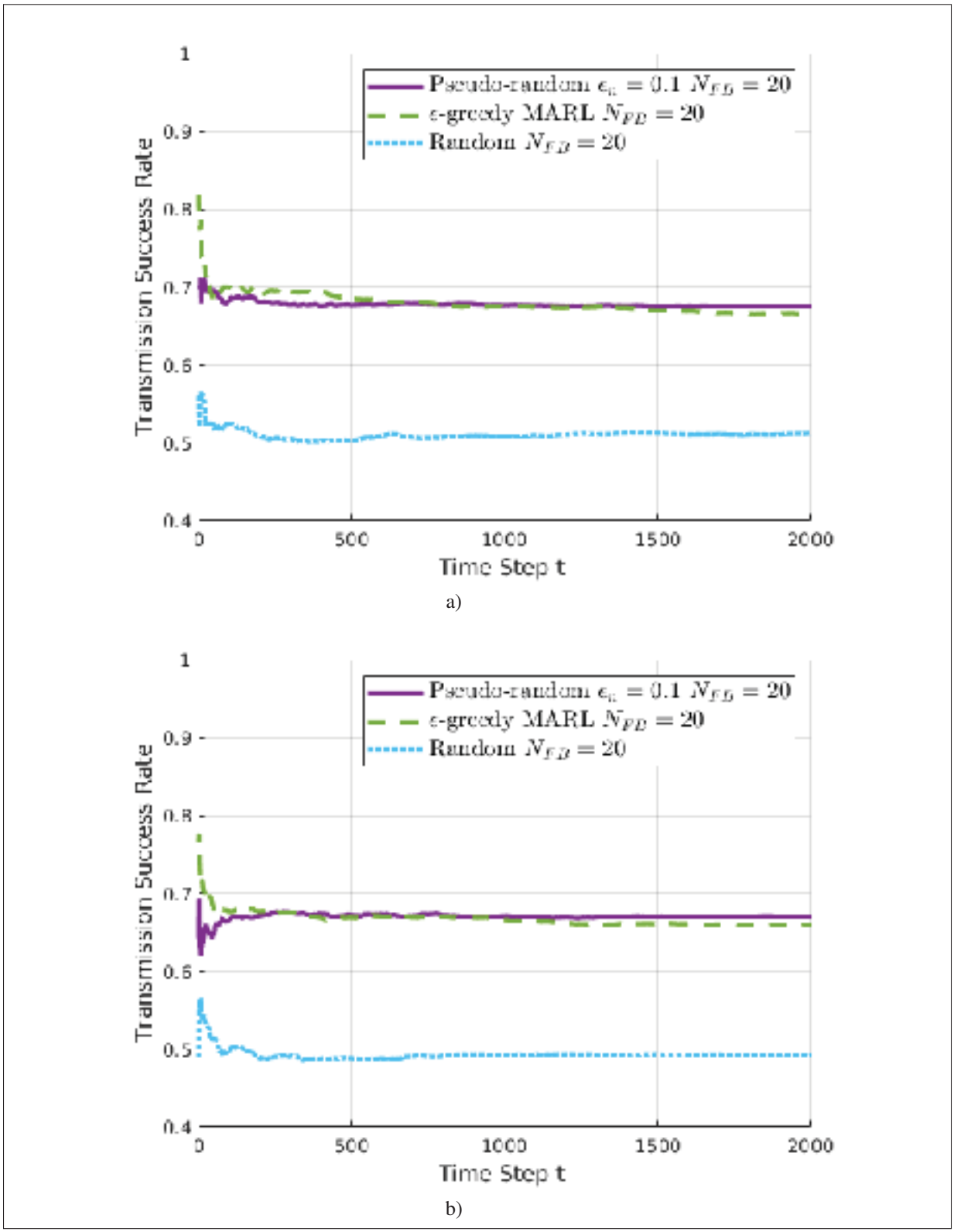


Figure 3.4 Performance evaluation using transmission success rate with 10 WNs and 20 channels using (a) local decision vectors and (b) super-decision vectors

added traffic overheads in terms of added time and bandwidth caused by each WN transmitting its local decision vector to each of its neighbouring nodes. The time overhead is especially important since by extending the collaboration phase of each time step, we reduce the amount of time available for the transmission phase.

Even though using the OR-rule could lead to an incorrect decision if even one observation gives rise to a false alarm, we observed that the detection probabilities $p_{d,m,AWGN}$ and $\bar{p}_{d,m,Ray}$ were generally higher than the false alarm probabilities $p_{fa,m}$, meaning that instances where a WN observation $\tau_t^i = occupied$ were more likely to be correct observations. Although this was generally true across our simulations, it was less so when running the simulation using Rayleigh fading, due to this model's higher probabilities of false alarm when the diversity order m is low. Nonetheless, we conclude that incorrect decisions due to false alarms were fairly infrequent.

3.5 Discussion

Based on the results obtained in the above section, we make the following assertions regarding the solution presented in this research project:

- Due to the unpredictable nature of the jammers, the collaborative pseudo-random channel selection algorithm gives a higher jammer detection ratio than an approach that uses multi-agent reinforcement learning to devise a transmission policy.
- Channel selection solutions based on reinforcement learning are suboptimal when the behaviour of the jammers cannot be learned and anticipated, to the extent that choosing these channels randomly results in a higher rate of detected jammers.
- When observations are especially unreliable (e.g. when assuming Rayleigh fading), it is more important for WNs to collaborate in order to increase the diversity order m of their observations than when observations are more reliable (as is the case of AWGN channels).

- Since using the collaborative pseudo-random channel selection algorithm (which leads to a higher JDR than the other two channel selection algorithms) also gives a higher TSR when modeling the channel using AWGN, it would appear as though improved jammer detection directly translates into safer transmissions across the network.

- Super-decision vectors, which allow WNs to obtain sensing information within a distance of two hops, lead to a higher rate of successful transmissions than local decision vectors, which only allow WNs to obtain information from their immediate neighbours.

3.6 Conclusion

Based on the results of our simulations, we deduce that the collaborative pseudo-random channel selection algorithm leads to improved awareness of which channels are under attack by jammers, demonstrated by a higher jammer detection ratio using this solution compared to MARL and random channel selection.

Additionally, synthesizing a node's sensing information with that of its neighbours into super-decision vectors in order to select channels for transmission leads to an improved rate of successful transmissions. The transmission success rate is further improved when super-decision vectors are used in conjunction with the pseudo-random channel selection algorithm.

CONCLUSION AND RECOMMENDATIONS

In this dissertation, we sought to offer a solution to the joint problem of anti-jamming and collaborative spectrum sensing in ad hoc tactical wireless networks. This challenge consists of developing strategies that members of these networks can use to sense the spectrum and share their observations with each other in order to improve their awareness of hostile jammers' activity. These strategies must also improve the rate of transmissions that occur on unjammed frequencies.

We represent the anti-jamming scenario as a partially observable stochastic game. This model simulates the interactions between a network of tactical communication systems, the frequency spectrum divided into a number of channels, and a group of hostile jammers. At each time step of the simulation, each jammer may be idle or actively jamming the channel it is assigned. We developed an algorithm where WNs can make use of their observations pertaining to the occupancy of a channel to decide which channel they will sense in the following time slot, while supporting their neighbours to mutually increase the accuracy of their observations. The objective of the WNs is to detect as many jamming instances as possible while also maximizing the number of transmissions performed on vacant channels. Results show that this collaborative pseudo-random channel algorithm allows the members of the network to detect jammers more frequently than a multi-agent reinforcement learning scheme found in the literature as well as randomly choosing which channel to sense.

Complementing our channel selection algorithm is the novel concept of super-decision vectors, where nodes share not only their local observations, but also the local decision vectors that they synthesize using their neighbours' observations. Members of the tactical wireless network can use these vectors to select a channel that they deem vacant and use it to transmit data. Simulations demonstrate that using super-decision vectors allows the nodes to more accurately determine which channels are vacant, which leads to a higher rate of transmissions occurring

on safe channels compared to only using nodes' local decision vectors to select transmission channels.

This work is important due to its military context. When military personnel is deployed in the field, ensuring proper communication between different groups of soldiers is essential not only to allow them to carry out their operations, but also to ensure their safety. Soldiers that are isolated because they are unable to communicate with their peers due to the presence of jammers are much more vulnerable, hence the importance of identifying jammers and transmitting on vacant channels.

In terms of future research, we recommend modifying the collaborative pseudo-random channel selection algorithm in order to dynamically adjust the level of inter-node collaboration that takes place during the sensing phase, based on the unreliability of the sensing action. This would allow the members of the network to increase the diversity order of their observations when sensing conditions are more prohibitive, for example when the channels undergo Rayleigh fading, thus resulting in more accurate observations in these conditions. Furthermore, steps should be taken to reduce the time and bandwidth overheads caused by using super-decision vectors. Reducing these constraints to manageable levels would allow this solution to be used in real-life tactical wireless networks.

APPENDIX I

COLLABORATIVE SPECTRUM SENSING IN TACTICAL WIRELESS NETWORKS

Bryan Gingras, Ali Pourranjbar, Georges Kaddoum

Department of Electrical Engineering, École de Technologie Supérieure,
1100 Notre-Dame Ouest, Montréal, Québec, Canada H3C 1K3

Article accepted for publication at the 54th IEEE International Conference on Communications (ICC 2020, Dublin, Ireland), June 2020.

1. Abstract

In this paper, we propose an algorithm for channel sensing, collaboration, and transmission for networks of Tactical Communication Systems that are facing intrusions from hostile Jammers. Members of the network begin by scanning the spectrum for Jammers, sharing this information with neighboring nodes, and merging their respective sets of observation data into a decision vector containing the believed occupancy of each channel. A user can then use this vector to find a vacant channel for its transmission. We introduce the concept of nodes sharing these vectors with their peers, who then merge them into super-decision vectors, allowing each node to better identify and select transmission channels. We consider fading scenarios that substantially limit the reliability of the users' observations, unless they cooperate to increase the trustworthiness of their sensing data. We propose a pseudo-random channel selection algorithm that strikes a balance between sensing reliability with the number of channels being sensed. Simulation results show that the proposed system improves the network's overall knowledge of the spectrum and the rate of Jammer-free transmissions while limiting added computational complexity to the nodes of the network, despite the Jammers' unpredictable nature.

2. Introduction

The security of wireless networks is a vital concern, particularly in wireless tactical communication networks that are subject to attacks from Jammers. Anti-jamming techniques have been

an important topic of research for many years, with jamming techniques evolving accordingly in an endless arms race. These attacks are particularly dangerous as they can degrade the performance of the network and possibly even cause denial of service (DoS). In a military context, a DoS attack can leave deployed personnel isolated and vulnerable. Despite the constant evolution of the technologies and the ever-increasing number of anti-jamming algorithms, the core principles behind anti-jamming remain the same, which consist primarily of avoiding Jammers or minimizing their ability to hinder transmissions within a network. Another approach relies on the detection of Jammers by sensing the power level at different frequencies in the electromagnetic spectrum in order to detect signals that may be of hostile origin (Grover *et al.* (2014)). Ensuring that tactical wireless networks remain free of the effects of hostile jammers requires a joint optimization of anti-jamming techniques as well as the mechanisms used by members of these networks to sense the spectrum.

Among recent research conducted in the field of anti-jamming, Slimeni *et al.* (2018) applied Q-learning to Secondary Users (SUs) in a Cognitive Radio Network (CRN) to teach them how long to transmit on each channel in the spectrum before it is visited by a Jammer. Thus, SUs do not need to continuously switch channels and can transmit on a channel for as long as the learned Jammer schedule allows. Zhang *et al.* (2018) used cooperative channel selection and power allocation to achieve multi-user and multi-channel anti-jamming. They accomplish this by sacrificing a fraction of some users' benefit to achieve higher overall system throughput, effectively forcing the Jammer into a situation where its ability to jam the network is limited.

Yao & Jia (2019) employed decentralized collaboration and multi-agent Q-learning to select channels on which to transmit, all while avoiding mutual interference as well as a sweeping Jammer. They also consider that the observations of their agents are imperfect due to sensing errors. Their solution's performance exceeds that of independent single-user Q-learning, and with a much faster convergence of the Q-table.

Aref & Jayaweera (2017) also used sweeping jamming and Q-learning in a multi-user setting to avoid mutual interference and a sweeping Jammer, but they did not employ inter-node collabo-

ration. Also, unlike Yao & Jia (2019), they maintained two separate Q-tables, one for selecting channels for sensing and another for transmission. Jia *et al.* (2018) considered a channel selection problem in dense wireless networks where the number of sensing agents varies over time. They proposed an anti-jamming dynamic game, and proved it to be an exact potential game, which guarantees the existence of at least one pure strategy Nash equilibrium (NE). Their approach, a "distributed anti-jamming channel selection algorithm", was employed, which leads to the NE of the anti-jamming game with multiple transmitters and multiple Jammers. This multi-agent learning algorithm runs iteratively until all sensing agents' sensing strategies (the probabilities of sensing each channel in the next iteration of the algorithm) converge to the NE.

Enhancing knowledge of the spectrum occupancy can improve the performance of anti-jamming techniques. The effect of sharing sensing information between agents is considered in Arshad & Moessner (2009) and Ghasemi & Sousa (2005). Arshad & Moessner (2009) use multi-agent collaboration in a CRN where observations are unreliable due to fading and adverse channel conditions. They demonstrate that multiple SUs sharing their sensing information with each other leads to significant gains in the detection probability when compared to local sensing. Ghasemi & Sousa (2005) address a similar topic, but they go further by exploring the impact of multi-agent collaboration in the case of spatially correlated shadowing, where nodes who are near each other experience similar shadowing effects. They show that nodes that are in close proximity to each other mutually degrade their performance and lower their probabilities of successful detection of a Primary User.

Although Slimeni *et al.* (2018), Zhang *et al.* (2018), Yao & Jia (2019), Aref & Jayaweera (2017) and Jia *et al.* (2018) propose solutions to the anti-jamming problem, they all consider Jammer behaviors as fully observable, which results in them being learnable and in some cases, predictable. Furthermore, there is clear potential in applying collaborative spectrum sensing and data fusion, such as in Arshad & Moessner (2009) and Ghasemi & Sousa (2005), to the particular context of tactical communications, which emphasizes anti-jamming and not only finding available bandwidth for transmission. Collaborative spectrum sensing, with an anti-

jamming problem where Jammer behavior cannot be predicted, is a topic of research that has not been adequately addressed.

In this paper, we propose an algorithm for channel selection, data collaboration, and transmission that leads to improved awareness of the spectrum as well as a higher rate of unjammed transmission in a partially observable environment where the behavior of the Jammer cannot be anticipated. We also explore new avenues in inter-node collaboration where each node in the network shares not only its local sensing data with its neighbors, but also its decisions with respect to the occupancy of each channel, thereby growing their respective set of observation data that they can use to select a vacant channel on which they may safely transmit. Finally, we introduce a collaborative channel selection algorithm that addresses unreliable sensing conditions. Our proposed system is thus an attempt to solve the joint problem of anti-jamming and collaborative spectrum sensing.

The remainder of the paper is organized as follows. Section 3 describes the system model and the simulated environment. Section 4 provides a detailed description of the approach we take to solve the anti-jamming problem. Section 5 presents the simulation setup and results while Section 7 concludes the paper.

3. System Model

In this work, an ad hoc network consisting of N_{WN} Wireless Nodes (WNs), N_{FB} orthogonal channels and a number of Jammers equal to the number of channels is considered. Each Jammer is assigned to one channel, and can only operate on it. Each WN possesses a transmission range, and any WN within this range is considered its neighbor, with which it will mutually share sensing information. Each WN periodically senses a channel on the spectrum to observe whether or not there is a Jammer operating on that particular channel. It then transmits this Jammer detection observation to its neighbors, resulting in each WN in the network possessing observations on one or more channels. Each WN then analyzes its set of observations using a fusion rule and deduces the occupancy of each channel in the spectrum, meaning that it will

decide whether each channel is being jammed or if it is safe to use for transmission. The WN then applies the channel selection algorithm to choose which channel it will sense next. Finally, the WN shares its occupancy decisions with its neighbors, selects a vacant channel and uses it to broadcast data. The WNs are dispersed geographically across a deserted terrain as displayed in Fig. 2.4.

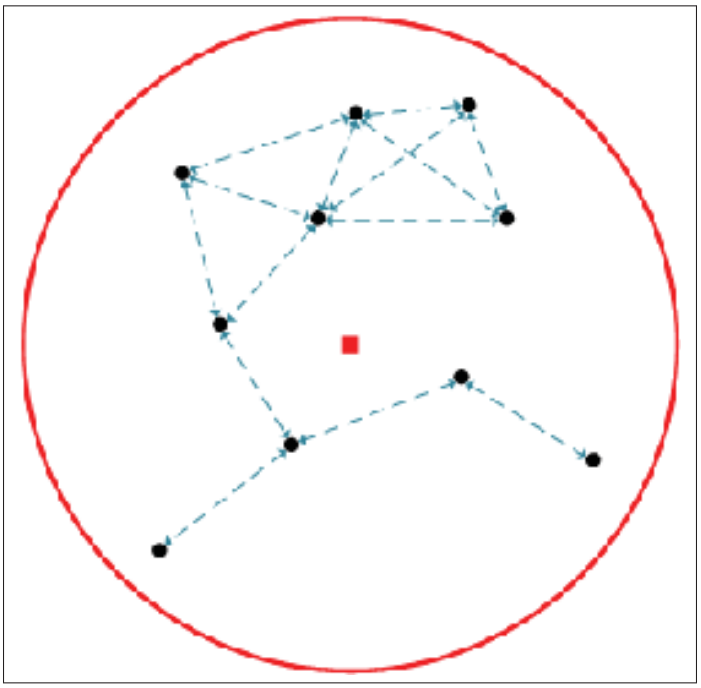


Figure-A I-1 Configuration of the network used in the simulation. The red box in the center represents the Jammers, the red circle represents their range of operation, the black circles represent the WNs, and the blue arrows between a pair of nodes indicate that they are neighbors and are able to share sensing information with each other

If a WN believes that all N_{WN} channels are jammed, it will not attempt to transmit. Additionally, channels for which the WN has no sensing information will be ignored by the WN, meaning that it will only attempt to transmit on channels which it has reason to believe is vacant. As in Lundén *et al.* (2013), we assume that the actions of each WN are sufficiently synchronized to allow all of their actions to occur simultaneously. We further assume that all agents remain active and immobile for the entirety of the simulation, and that the number of

WNs and Jammers is fixed. Lastly, nodes share their sensing information using a common control channel which we assume cannot be jammed nor subject to collisions (Lo & Akyildiz (2012)).

Observations made by the WNs are not perfect and depend on probabilities of detection when the observed channel is jammed, as well as probabilities of false alarm when the observed channel is vacant (meaning the WN will falsely detect a Jammer). Furthermore, observations are subject to signal fading. Particularly, we separately consider Additive White Gaussian Noise (AWGN) channels as well as Rayleigh fading, both of which differently affect the probability that a WN's observation will be incorrect. Due to the multi-path fading component of the Rayleigh model, its impact on the sensing reliability is higher than that of the AWGN model. These probabilities vary with respect to the number of WNs simultaneously sensing the channel, meaning that observations are more likely to be correct if multiple WNs sense a given channel at the same time. When using AWGN channels, in addition to the number of users m simultaneously sensing a channel, probabilities of successful detection $p_{d,m}$ are also a function of the SNR γ of the received signal, as shown in (A I-1). The Rayleigh fading model considers that the WN samples the signal N times, which leads to an average detection probability $\bar{p}_{d,Ray}$ as shown in (A I-2). Equations (A I-1) and (A I-2) are detailed in Digham *et al.* (2003). Rayleigh detection probabilities with $m > 1$ are given by (A I-3). In all cases, σ^2 is the variance of the sampled signal, a is a non centrality parameter, and λ is a decision threshold.

$$p_{d,m,AWGN} = Q_{mN/2}\left(\sqrt{\frac{a\gamma}{\sigma^2}}, \sqrt{\frac{\lambda}{\sigma^2}}\right), \tag{A I-1}$$

$$\bar{p}_{d,Ray} = e^{-\frac{\lambda}{2\sigma^2}} \sum_{i=0}^{N/2-2} \frac{\left(\frac{\lambda}{2\sigma^2}\right)^i}{i!} + \left(\frac{2\sigma^2 + a\bar{\gamma}}{a\bar{\gamma}}\right)^{N/2-1} \times \left(e^{-\frac{\lambda}{2\sigma^2+a\bar{\gamma}}} - e^{-\frac{\lambda}{2\sigma^2}} \sum_{i=0}^{N/2-2} \frac{\left(\frac{\lambda a\bar{\gamma}}{2\sigma^2(2\sigma^2+a\bar{\gamma})}\right)^i}{i!}\right), \tag{A I-2}$$

$$\bar{p}_{d,m,Ray} = 1 - \prod_{i=1}^{m}(1 - \bar{p}_{d,Ray,i}). \tag{A I-3}$$

The power of the Jammer signal received by the user is given by $P_T \times (d/d_0)^{\phi}$, where d/d_0 is the distance between the transmitter and the receiver divided by a reference distance, the constant ϕ corresponds to an attenuation factor which depends on the physical environment in which the transmission takes place, and P_T represents the initial power of the transmitted signal. We consider this received power to include noise as well as the channel gain in the case of Rayleigh fading. WNs must therefore cooperate in order to increase the diversity order m and obtain more reliable sensing information.

At any given moment, a Jammer may be idle, or it may be actively jamming the channel in order to intercept transmissions between WNs. We represent its behavior using a Markov model with states 0 and 1, which respectively signify idle and active. We define $p_{k,00}$ as the probability that, at each time step, Jammer k remains in the idle state, and $p_{k,11}$ as the probability that it stays in the active state. It follows that $p_{k,01} = (1 - p_{k,00})$ is the probability that the Jammer goes from idle to active, and that $p_{k,10} = (1 - p_{k,11})$ is the probability of going from active to idle. The corresponding Markov model is represented by Fig. I-2.

4. Stochastic Game Formulation

Each WN can only sense one channel at a time, therefore making it impossible for it to observe the entire spectrum at once. The multi-agent jamming problem can be represented as a partially observable stochastic game modelled as follows:

- A set of time steps $t = 1, 2, \ldots, T$ that are each of fixed length and split into three sub-slots: sensing, collaboration, and transmission.
- A group of Wireless Nodes $M = \{1, \ldots, N_{WN}\}$ dispersed throughout the environment.

- A group of N_{FB} channels in the frequency band. Each WN senses exactly one of these channels at each time step t and observes its occupancy.

- A set of possible occupancy values for a channel j as it would be sensed by WN i at time step t, given by $s_t^{i,j} \in \{vacant, occupied\}$.

- The vector $\boldsymbol{s}_t^i$ represents the occupancy of each channel as it would be seen by WN i at time step t, given by $\boldsymbol{s}_t^i = [s_t^{i,1}, \ldots, s_t^{i,N_{FB}}]$. Its value is a member of the set $\{vacant, occupied\}$.

- The state $\boldsymbol{s}_t$ is an array of vectors $\boldsymbol{s}_t^i$, showing each WN's perception of each channel at time t, given by $\boldsymbol{s}_t = [\boldsymbol{s}_t^1, \ldots, \boldsymbol{s}_t^{N_{WN}}]$.

- A set of possible sensing actions $a_t^i \in A_i$, where $i \in M$, and $A_i = \{1, \ldots, N_{FB}\}$ represents which channel will be sensed by the WN at time step t.

- τ_t^i represents an observation made by WN i at time t, i.e. it is the observation made from taking action a_t^i. Its value is a member of the set $\{vacant, occupied\}$. Due to the imperfect nature of the WNs' observations, this value may not reflect the actual occupancy of the channel as given by $s_t^{i,j}$.

- A vector of sensing decisions made by each WN i $\boldsymbol{d}_t^i \in \boldsymbol{D}_{i,t}$. Each element of $\boldsymbol{d}_t^i$ is a member of the set $\{vacant, occupied\}$.

- A vector of super-decision vectors $\boldsymbol{\delta}_t^i$ obtained by combining WN i's decision vector $\boldsymbol{d}_t^i$ with its neighbors' decision vectors $\boldsymbol{d}_t^j \ \forall \ j$ into a single vector. Like $\boldsymbol{d}_t^i$, each element of $\boldsymbol{\delta}_t^i$ is a member of the set $\{vacant, occupied\}$.

- A transmission outcome $x_t^i \in \{successful, jammed\}$ representing the outcome of the transmission performed by WN i at time step t.

- A state transition function ϕ: $\boldsymbol{S} \times \boldsymbol{A} \times \boldsymbol{S} \rightarrow \mathbb{R}$ that defines the state transition probabilities $P(\boldsymbol{s}_{t+1} \mid \boldsymbol{s}_t, a_t^1, \ldots, a_t^{N_{WN}})$. We use the Jammer probabilities $p_{k,00}$ and $p_{k,11}$ as the state transition probabilities. We assume that the agents' actions do not affect the state transition probabilities: $P(\boldsymbol{s}_{t+1} \mid \boldsymbol{s}_t, a_t^1, \ldots, a_t^{N_{WN}}) = P(\boldsymbol{s}_{t+1} \mid \boldsymbol{s}_t)$. In other words, we assume that an

WN's sensing action does not cause any interference on the channel that could affect the observation of another WN.

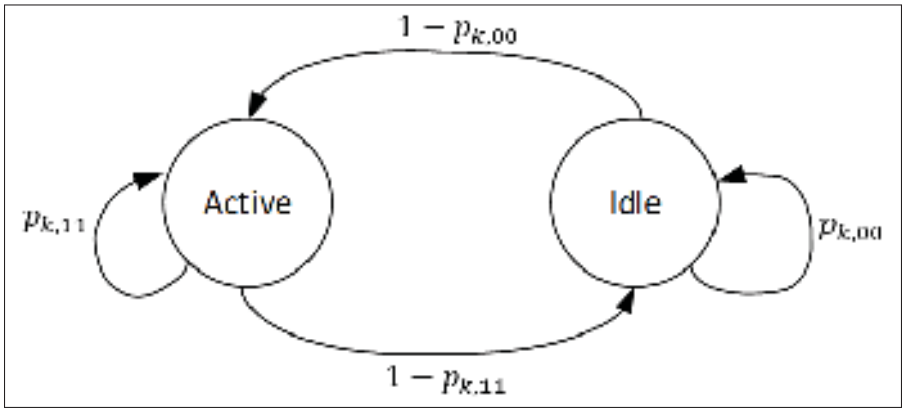


Figure-A I-2 Two-State Jammer Markov Model. Upon initialization, a Jammer's initial state is decided randomly. At each time step, it may remain in its current state according to probabilities $p_{k,00}$ and $p_{k,11}$ or switch to the alternate state

4.1 Spectrum Sensing, Cooperation and Fusion

A time step of the simulation can be summarized by the following. Each WN i senses a channel and receives a sensing observation datum τ_t^i, which it then shares with each of its neighboring WNs j. Each WN i then combines its observations into a decision vector $\boldsymbol{d}_t^i$ by applying a fusion rule on its set of observation data. Then, basing itself on the outcome of the observation that it has made, as well as the actions taken in the previous time step by its neighbors, each WN chooses which channel to sense in the following time step $t+1$. Next, in order to improve their knowledge of the presence of Jammers in the spectrum, the WNs transmit their respective decision vectors $\boldsymbol{d}_t^i$ to their neighbors, who again apply a fusion rule, resulting in a super-decision vector $\boldsymbol{\delta}_t^i$. In doing so, an WN i not only synthesizes the observations made by itself and its neighbors j, but also gains indirect access to the observations made by j's neighbors that are outside i's transmission range, thereby increasing the ranged of shared information from one hop to two. Finally, each node uses $\boldsymbol{\delta}_t^i$ to select a channel it believes to be vacant and transmits on it. This is summarized in Algorithm 1.

Algorithm-A I-1 Sensing, cooperation, and fusion algorithm

Initialize $t = 0$
Initialize set of Wireless Nodes M
Initialize set of Jammers J
Initialize set of Channels W
Initialize $a_0^i = random(1,\ldots,N_{FB}),\ i \in \{1,\ldots,N_{WN}\}$
Initialize $w_0^z = bernoulli(0.5),\ z \in \{1,\ldots,N_{FB}\}$
$\boldsymbol{s}_0 = computeOccupancy(M,J,W)$
while $t < T$ **do**
    // Sensing phase
    $\tau_t^i = sense(a_t^i),\ i \in \{1,\ldots,N_{WN}\}$
    // Collaboration phase
    for *each WN i in M* **do**
        Transmit tuple $\{a_t^i, \tau_t^i\}$ to neighbours
        for *each neighbour j of WN i* **do**
            j receives $\{a_t^i, \tau_t^i\}$ from i
            j adds a_{t+1}^i to its action vector: $\boldsymbol{a}_{t+1}^j = \boldsymbol{a}_{t+1}^j \cup a_{t+1}^i$
        end
    end
    $\boldsymbol{d}_t^i = fusion(\tau_t^i, \{\tau_t^j\}_{\forall j}), i \in \{1,\ldots,N_{WN}\}$
    $a_{t+1}^i = chooseAction(\tau_t^i, \boldsymbol{a}_t^i), i \in \{1,\ldots,N_{WN}\}$
    for *each WN i in M* **do**
        Transmit decision vector $\boldsymbol{d}_t^i$ to neighbours
        for *each neighbour j of WN i* **do**
            j receives $\boldsymbol{d}_t^i$ from i
        end
        $\boldsymbol{\delta}_t^i = fusion(\boldsymbol{d}_t^i, \{\boldsymbol{d}_t^j\}_{\forall j})$
        // Transmission Phase
        $x_t^i = transmit(\boldsymbol{\delta}_t^i)$
    end
    $\boldsymbol{s}_{t+1} = computeOccupancy(M,J,W)$
    $t = t + 1$
end

We repurpose the transmission sub-time slot selection algorithm from Lundén *et al.* (2015) for our sensing channel selection algorithm. The algorithm functions as follows: if a WN detects a Jammer on a given channel, it will sense that channel again during the following time slot $t+1$. If the sensing action does not detect a Jammer, then we enter an exploration-exploitation

scenario, where there is a probability $\varepsilon_n \in [0,1]$ that the WN will exploit its neighbors' knowledge of the spectrum by selecting the action a_t^j of one of its neighbors j. If taking action a_t^j yields a Jammer, then j is certain to choose this action again for the next time slot $t+1$, and if WN i chooses this action as well for the same time slot, then i and j assist each other by increasing the diversity order m and therefore the detection probability $p_{d,m}$ of both WNs. If the observation τ_t^j is a false positive, then a repeated observation on that channel with a higher value of m (and therefore a lower probability of false alarm) will be more likely to yield the correct observation of *vacant*.

If the two previous criteria are not met, the third possibility is for the WN to select a channel that was not sensed by itself or by any of its neighbors, in order to explore a greater range of channels that are not currently being sensed by the WN or its surrounding nodes. The value of ε_n is therefore a trade-off between sensing reliability and the number of channels being sensed at any given moment. This is summarized in Algorithm 2.

Algorithm-A I-2 Collaborative pseudo-random channel selection algorithm

```
if τ_t^i == occupied then
    a_{t+1}^i = a_t^i
end
else
    if u(0,1) ≤ ε_n then
        a_{t+1}^i = a_t^j
    end
    else
        a_{t+1}^i = random(A_i \ {a_t^i, a_t^j ∀ j})
    end
end
```

The fusion strategy used by the WNs to merge their sensing information into a decision vector $\boldsymbol{d}_t^i$ is the OR rule, meaning that each WN considers a channel to be jammed if its own observation, or one that it receives from at least one of its neighbors, indicates that the channel is

jammed. The WNs apply this same OR fusion rule when combining their decision vectors into a super-decision vector $\boldsymbol{\delta}_t^i$.

5. System Performance

Since the major contribution of this article lies in the algorithm used for action selection, we judge the effectiveness of our solution by comparing the performance of this algorithm against a collaborative multi-agent reinforcement learning scheme that attempts to build a channel selection policy based on sensing data as proposed in Lundén *et al.* (2013). Additionally, we compare our results with those obtained using randomly chosen actions.

We evaluated the performance of the algorithms described above using a simulation of a deserted, empty terrain containing a set number of WNs and Jammers implemented in Python 3.5.2. The simulation ran for a pre-set number of time steps T according to the stochastic game described above. Throughout the simulation, we computed the number of detected incidences of jamming over the total number of times that jamming has occurred up until the current time step. As a secondary performance metric, we also calculated the number of successful transmissions over the total number of transmissions that had been attempted up to and including the current time step. Since each WN attempts to transmit once per time step, we knew this latter value ahead of time to be simply $N_{WN} \times t$.

The scenario used for evaluation consisted of 10 WNs and 10 Jammers (and therefore 10 channels). For each Jammer, the values of $p_{k,00}$ and $p_{k,11}$ were randomly generated between bounds of 0.85 and 0.98 as in Lundén *et al.* (2013). Due to the high values of these probabilities, a Jammer k was much more likely to remain in its current state from one time step to the next than to transition to the alternate state. The simulation was run for $T = 2000$ time steps. Transmitted signals were attenuated at a rate of $P_T \times (d/0.05)^{-2.3}$. We used the exponent -2.3 to represent a flat, empty environment much like a desert or field. The WNs were arranged across the terrain as per Fig. I-1. We also used a reference distance of 0.05 *km* Lundén *et al.* (2013). We used the following false alarm probabilities: $p_{fa,1} = 0.0015$, and $p_{fa,n} = 10^{-7}$ for $n \geq 2$

for AWGN channels, as well as $p_{fa,1} = 0.83$, $p_{fa,2} = 0.32$, $p_{fa,3} = 0.03$, $p_{fa,4} = 0.003$, and $p_{fa,n} = 0.001$ for $n \geq 5$ Lundén *et al.* (2013) for Rayleigh fading channels.

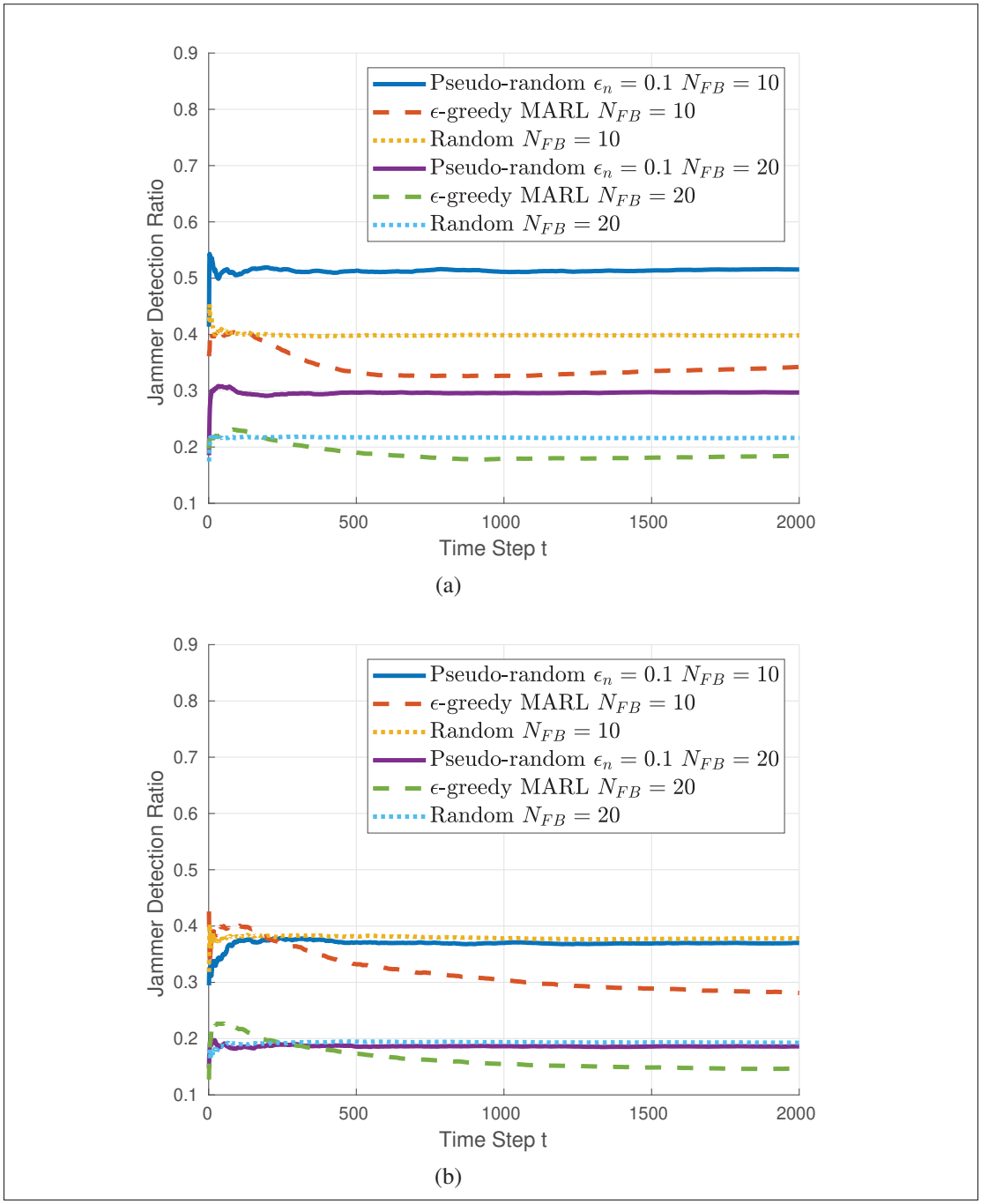


Figure-A I-3 Performance evaluation using jammer detection ratio with (a) AWGN channels and (b) Rayleigh fading

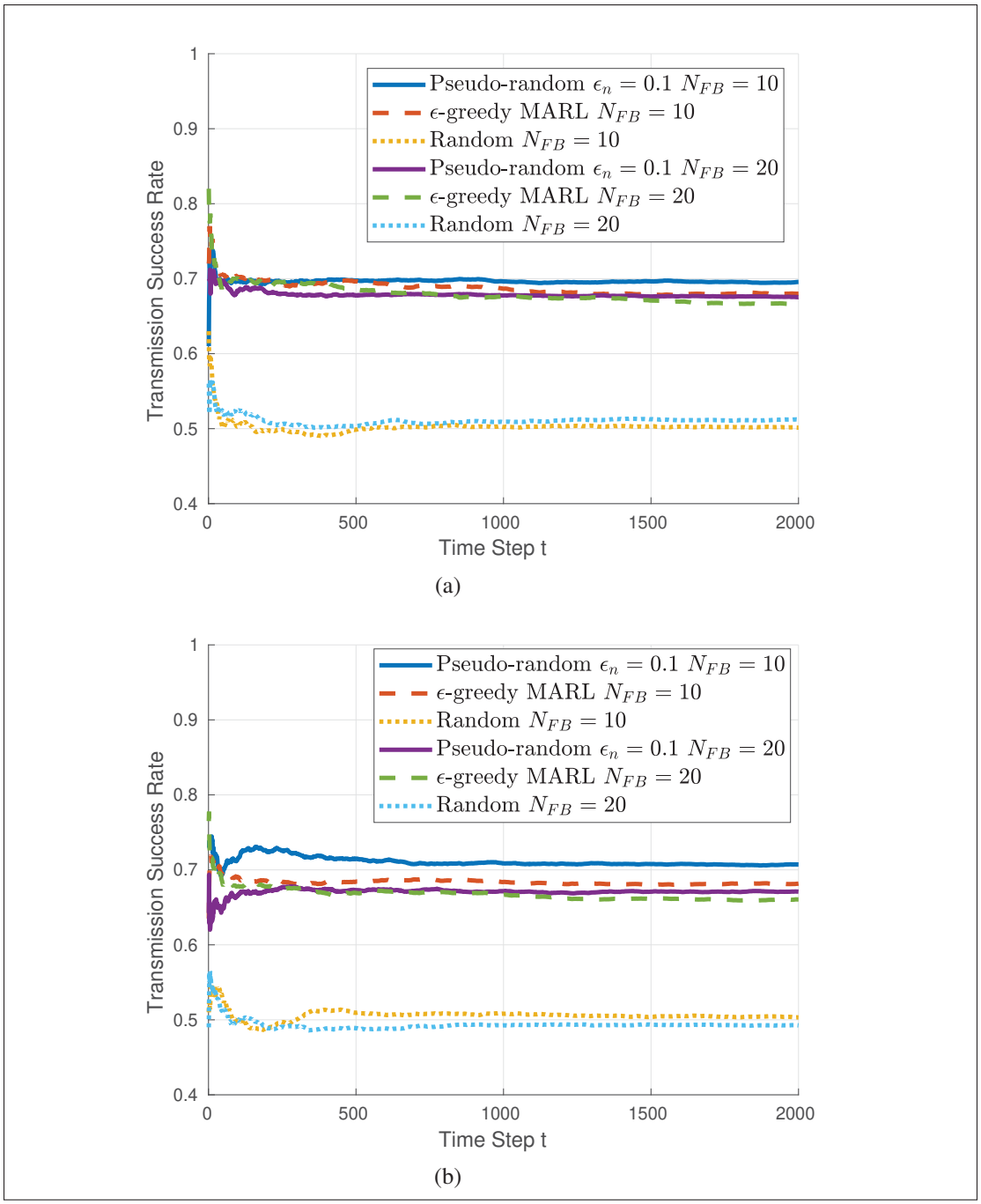


Figure-A I-4 Performance evaluation using transmission success rate with (a) local decision vectors and (b) super-decision vectors

The Jammers' transmission power was held constant at 15 dB. Using values of received SNR by the WNs ranging from 0 dB to 15 dB and diversity orders ranging from 1 to 6 WNs simul-

taneously sensing the same channel, we used (A I-1) to compute a 2D matrix that served as a look-up table for the different values of $p_{d,m,AWGN}$ that the simulation software could consult every time a WN sensed a channel. Similarly, we used (A I-2) to compute a 1D array giving detection probabilities $\bar{p}_{d,Ray}$ where $m = 1$, using the same range of SNR values and when using the Rayleigh model for channel fading. Detection probabilities with Rayleigh fading when $m > 1$ were dynamically calculated during the simulation using (A I-3). The variance σ^2 is equal to 1, a is set to 2, and λ is equal to 12.1 in order to approximate probabilities seen in Lundén *et al.* (2013).

Since we used an OR fusion rule, a single false alarm could lead to an incorrect decision when using this strategy, but the false alarm probabilities used in this simulation were generally smaller than the values of $p_{d,m,AWGN}$ and $\bar{p}_{d,m,Ray}$ that we generally observed, which means that most observations where τ_t^i was equal to *occupied* were due to correct observations and not false alarms, though it is important to note that observations are less reliable when using Rayleigh fading and were thus more prone to incorrect decisions than when using AWGN channels.

In Figs. I-3a and I-3b we illustrate the performance of our algorithm with $\varepsilon_n = 0.1$ as well as the multi-agent reinforcement learning scheme and random action selection using AWGN channels and Rayleigh fading, respectively. The probabilities of a false alarm were higher for Rayleigh fading, which explains why the algorithms generally show lower performance in Fig. I-3b than in Fig. I-3a.

Next, in Figs. I-4a and I-4b, we observe the rate of successful transmission, i.e. the number of transmissions that took place on a vacant channel over the total number of attempted transmissions $N_{WN} \times t$, using simple decision vectors $\boldsymbol{d}_t^i$ versus super-decision vectors $\boldsymbol{\delta}_t^i$. Furthermore, all of the simulations used to determine the transmission success rate were done using AWGN channels. Time steps where the WN could not find a vacant channel for transmission were excluded from the calculation. Each curve in each figure is an average of 100 iterations of the simulation.

6. Jammer Detection

We remark in Fig. I-3a that the pseudo-random algorithm out-performed the multi-agent reinforcement learning scheme and random action selection. The algorithm exploited the fact that jammed channels were likely to remain jammed, and would continue to sense the channel until it suddenly became vacant. In such an event, the WN could, according to ε_n, assist its neighbor or sense an unsensed channel in the hopes of finding a new channel that it could exploit over several time steps. In I-3b, the performance of the pseudo-random algorithm is near that of the random action selection algorithm. The higher likelihood of incorrect observations when using Rayleigh can give rise to a situation where the WN suddenly senses a jammed channel as being vacant, which causes it to stop sensing that channel and sense a different, potentially vacant channel.

Due to the random nature of the Markov model controlling the Jammers' state transitions, it is impossible to anticipate when a Jammer will become idle or active. The inherent randomness of the Jammer may explain how random action selection performs as well as it does. This also explains the MARL algorithm's difficulty to devise a useful transmission policy. When we increased N_{FB} from 10 to 20, we essentially doubled the number of possible instances of jamming that could occur during the simulation, and considering that the number of WNs remains the same, we observe the expected result which is for the performance of each algorithm to drop by half when doubling N_{FB}.

6.1 Transmission Success Rate

In Fig. I-4a, we can observe that the relative performance of each algorithm is fairly similar to that of Fig. I-4b in terms of the rate of successful transmission. We also notice that the performance of an algorithm does not vary considerably when $N_{FB} = 10$ versus $N_{FB} = 20$. This can be attributed to the fact that a WN only transmits on channels for which it possesses information on its occupancy. In other words, even if the number of channels for which the WN has no information increases, the number of sensing data that it possesses does not change

due to the fact that N_{WN} remains constant at 10. However, the average diversity order of each sensing action may be reduced if N_{FB} is raised to 20, resulting in less reliable observations, which may explain why the cases where $N_{FB} = 10$ generally perform slightly better than when $N_{FB} = 20$. Therefore, increasing N_{FB} beyond N_{WN} does not vastly affect the transmission success rate.

We note that in both figures, the performance roughly does not exceed 70%, which may be an asymptotic value that is a function of N_{WN} as well as other factors including the values of the probabilities of false alarm. Despite this, we see that using super-decision vectors $\boldsymbol{\delta}_t^i$ performs slightly better than when using local decision vectors. However, this simulation does not consider the added overheads, both in terms of time and bandwidth, that arises when every WN shares its local decision vector $\boldsymbol{d}_t^i$ with its neighbors. Particularly, it is important to note that extending the collaboration phase of each time step reduces the amount of time available for the WNs to transmit data, which would be a crucial consideration if this simulation modeled time as being continuous instead of a succession of time slots. Lastly, it is interesting to note that despite the fact that the system model is geared towards improving the Jammer detection ratio, our algorithm still leads to an improved transmission success rate, which gives credence to the idea that improved Jammer detection translates into safer transmissions.

7. Conclusion

In this article, we demonstrated the effectiveness of a simple, yet promising, pseudo-random channel sensing algorithm for tactical wireless networks along with an innovative data collaboration scheme that makes better use of neighboring nodes' sensing information using super-decision vectors. Awareness of spectrum usage favors Jammer-free transmissions between Wireless Nodes, which is an important consideration for ensuring the safety and effectiveness of military personnel on the field. When coupled with appropriate data fusion algorithms, this algorithm leads to a higher performance with respect to the rate of Jammer detection and the number of non intercepted transmissions compared to random channel selection as well as multi-agent reinforcement learning.

BIBLIOGRAPHY

Ahmed, N., Hadaller, D. & Keshav, S. (2006). GUESS: gossiping updates for efficient spectrum sensing. *Proceedings of the 1st international workshop on Decentralized resource sharing in mobile computing and networking*, pp. 12–17.

Akyildiz, I. F., Lee, W.-Y., Vuran, M. C. & Mohanty, S. (2006). NeXt generation/dynamic spectrum access/cognitive radio wireless networks: A survey. *Computer networks*, 50(13), 2127–2159.

Aref, M. A. & Jayaweera, S. K. (2017). A novel cognitive anti-jamming stochastic game. *2017 Cognitive Communications for Aerospace Applications Workshop (CCAA)*, pp. 1–4.

Arshad, K. & Moessner, K. (2009). Collaborative spectrum sensing for cognitive radio. *Proc. IEEE International Conference on Communications (ICC)*, pp. 1–5.

Awan, F., Sheikh, N. & Muhammad, F. (2009). Outer bounds for the symmetric gaussian cognitive radio channel with dpc encoded cognitive transmitter. *Proceedings of the World Congress on Engineering*, 1.

Bhattacharya, B. & Saha, B. (2014). Community Model Architecture–A New Data Fusion Paradigm for Implementation. *International Journal of Innovative Research in Computer and Commmucation Engineering*, 2(6), 4774–4783.

Bickenbach, F. & Bode, E. (2003). Evaluating the Markov property in studies of economic convergence. *International Regional Science Review*, 26(3), 363–392.

Bonanno, G. (2018). *Game Theory: Volume 1: Basic Concepts*. CreateSpace Independent Publishing Platform. Consulted at https://books.google.ca/books?id=bI-UswEACAAJ.

Broustis, I., Pelechrinis, K., Syrivelis, D., Krishnamurthy, S. V. & Tassiulas, L. (2009). FIJI: Fighting implicit jamming in 802.11 WLANs. *International Conference on Security and Privacy in Communication Systems*, pp. 21–40.

Cai, Y., Mo, Y., Ota, K., Luo, C., Dong, M. & Yang, L. T. (2014). Optimal data fusion of collaborative spectrum sensing under attack in cognitive radio networks. *IEEE Network*, 28(1), 17–23.

Castanedo, F. (2013). A review of data fusion techniques. *The Scientific World Journal*, 2013.

Daily, D. I. (2005). Field Report: Putting the ICE on IEDs. Accessed: 2019-11-15, Consulted at https://www.defenseindustrydaily.com/field-report-putting-the-ice-on-ieds-0916/.

Digham, F. F., Alouini, M.-S. & Simon, M. K. (2003). On the energy detection of unknown signals over fading channels. *IEEE International Conference on Communications, 2003. ICC'03.*, 5, 3575–3579.

Ericsson. (2017). Ericsson Internet of Things Forecast. Accessed: 2019-11-11, Consulted at https://www.ericsson.com/en/mobility-report/internet-of-things-forecast.

Fu, Y., He, Z. & Yang, F. A. N. (2017). A simple quantization-based multibit cooperative spectrum sensing for cognitive radio networks. *14th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, pp. 220–223.

Ghasemi, A. & Sousa, E. S. (2005). Collaborative spectrum sensing for opportunistic access in fading environments. *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005.*, pp. 131–136.

Grover, K., Lim, A. & Yang, Q. (2014). Jamming and anti-jamming techniques in wireless networks: a survey. *International Journal of Ad Hoc and Ubiquitous Computing*, 17(4), 197–215.

Harris. (2019). Electronic Warfare | Harris. Accessed: 2019-11-20, Consulted at https://www.harris.com/content/electronic-warfare.

IEEE. (2010). IEEE Draft Standard for Part 22 . 1 : Standard to Enhance Harmful Interference Protection for Low Power Licensed Devices Operating in TV Broadcast Bands. *IEEE Std 802.22.1-2010*.

Jia, L., Xu, Y., Sun, Y., Feng, S., Yu, L. & Anpalagan, A. (2018). A game-theoretic learning approach for anti-jamming dynamic spectrum access in dense wireless networks. *IEEE Trans. Veh. Technol.*, 68(2), 1646–1656.

Kang, X., Liang, Y. C., Nallanathan, A., Garg, H. K. & Zhang, R. (2009). Optimal power allocation for fading channels in cognitive radio networks: Ergodic capacity and outage capacity. *IEEE Transactions on Wireless Communications*, 8(2), 940–950.

Kolodzy, P. (2002). *FCC Spectrum policy task force report*.

Kusaladharma, S. & Tellambura, C. (1999). An overview of cognitive radio networks. *Wiley Encyclopedia of Electrical and Electronics Engineering*, 1–17.

Lazos, L., Liu, S. & Krunz, M. (2009). Mitigating control-channel jamming attacks in multi-channel ad hoc networks. *Proceedings of the second ACM conference on Wireless network security*, pp. 169–180.

Lien, Y.-N., Jang, H.-C. & Tsai, T.-C. (2009). A MANET based emergency communication and information system for catastrophic natural disasters. *2009 29th IEEE International Conference on Distributed Computing Systems Workshops*, pp. 412–417.

Lo, B. F. & Akyildiz, I. F. (2012). Multiagent jamming-resilient control channel game for cognitive radio ad hoc networks. *2012 IEEE International Conference on Communications (ICC)*, pp. 1821–1826.

Lundén, J., Kulkarni, S. R., Koivunen, V. & Poor, H. V. (2013). Multiagent reinforcement learning based spectrum sensing policies for cognitive radio networks. *IEEE Journal of Selected Topics in Signal Processing*, 7(5), 858–868.

Lundén, J., Motani, M. & Poor, H. V. (2015). Distributed algorithms for sharing spectrum sensing information in cognitive radio networks. *IEEE Transactions on Wireless Communications*, 14(8), 4667–4678.

Mpitziopoulos, A., Gavalas, D., Pantziou, G. & Konstantopoulos, C. (2007). Defending wireless sensor networks from jamming attacks. *2007 IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 1–5.

Muraleedharan, R. & Osadciw, L. A. (2006). Jamming attack detection and countermeasures in wireless sensor network using ant system. *Wireless Sensing and Processing*, 6248, 62480G.

Musavian, L., Aïssa, S. & Member, S. (2009). Fundamental Capacity Limits of Cognitive Radio in Fading Environments with Imperfect Channel Information. *IEEE Transactions on Communications*, 57(11), 3472–3480. doi: 10.1109/TCOMM.2009.11.070410.

Pelechrinis, K., Iliofotou, M. & Krishnamurthy, S. V. (2010). Denial of service attacks in wireless networks: The case of jammers. *IEEE Communications surveys & tutorials*, 13(2), 245–257.

Renault, J. (2019). A tutorial on Zero-sum Stochastic Games. *arXiv preprint 1905.06577*.

Roozgard, A., Tachwali, Y., Barzigar, N. & Cheng, S. (2012). Collaborative Spectrum Sensing for Cognitive Radio Networks. *Foundation of Cognitive Radio Systems*, 97.

Sahai, A., Hoven, N. & Tandra, R. (2004). Some fundamental limits on cognitive radio. *Allerton Conference on Communication, Control, and Computing*, pp. 1662–1671.

Sen, R. et al. (2012). Distributed wireless communications for tactical network dominance. Google Patents. US Patent 8,254,847.

Seo, J., Chen, Y.-H., De Lorenzo, D. S., Lo, S., Enge, P., Akos, D. & Lee, J. (2011). A real-time capable software-defined receiver using GPU for adaptive anti-jam GPS sensors. *Sensors*, 11(9), 8966–8991.

Shapley, L. S. (1953). Stochastic games. *Proceedings of the national academy of sciences*, 39(10), 1095–1100.

Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., Lanctot, M., Sifre, L., Kumaran, D., Graepel, T. et al. (2018). A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science*, 362(6419), 1140–1144.

Slimeni, F., Scheers, B., Chtourou, Z., Nir, V. L. & Attia, R. (2018). A modified Q-learning algorithm to solve cognitive radio jamming attack. *International Journal of Embedded Systems*, 10(1), 41–51.

Solan, E. & Vieille, N. (2015). Stochastic games. *Proceedings of the National Academy of Sciences of the United States of America*, 112(45), 13743–13746.

Son, K., Chul, B. & Song, J. (2013). Power allocation policies with full and partial inter-system channel state information for cognitive radio networks. 99–113. doi: 10.1007/s11276-012-0453-0.

Stamp, M. (2004). A revealing introduction to hidden Markov models. *Department of Computer Science San Jose State University*, 26–56.

Staple, G. & Werbach, K. (2004). IEEE Spectrum: the end of spectrum scarcity. Accessed: Accessed: 2019-11-20, Consulted at spectrum.ieee.org/telecom/wireless/the-end-of-spectrum-scarcity.

Stevenson, C. R., Chouinard, G., Lei, Z., Hu, W., Shellhammer, S. J. & Caldwell, W. (2009). IEEE 802.22: The first cognitive radio wireless regional area network standard. *IEEE communications magazine*, 47(1), 130–138.

Strasser, M., Danev, B. & Čapkun, S. (2010). Detection of reactive jamming in sensor networks. *ACM Transactions on Sensor Networks (TOSN)*, 7(2), 16.

Sutton, R. S. & Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.

Tilghman, P. (2016). DARPA Spectrum Collaboration Challenge (SC2). Accessed: 2019-11-11, Consulted at https://www.darpa.mil/program/spectrum-collaboration-challenge.

Visotsky, E., Kuffner, S. & Peterson, R. (2005). On collaborative detection of TV transmissions in support of dynamic spectrum sharing. *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005.*, pp. 338–345.

Wang, W. (2009). A Brief Survey on Cognitive Radio. In *Cognitive Radio Systems*. IntechOpen.

Wang, Y., Li, H. & Qian, L. (2013). Belief propagation based spectrum sensing subject to dynamic primary user activities: Phantom of quickest detection. *MILCOM 2013-2013 IEEE Military Communications Conference*, pp. 1193–1200.

Watkins, C. J. C. H. (1989). *Learning from delayed rewards*. (Ph.D. thesis, King's College, Cambridge, King's Parade, Cambridge CB2 1ST, United Kingdom).

Wilson, C. (2007). Improvised Explosive Devices (IEDs) in Iraq and Afghanistan: Effects and Countermeasures. Accessed: 2019-11-15, Consulted at https://www.defenseindustrydaily.com/field-report-putting-the-ice-on-ieds-0916/.

Xu, W., Trappe, W., Zhang, Y. & Wood, T. (2005). The feasibility of launching and detecting jamming attacks in wireless networks. *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*, pp. 46–57.

Yao, F. & Jia, L. (2019). A Collaborative Multi-agent Reinforcement Learning Anti-jamming Algorithm in Wireless Networks. *IEEE Wireless Communications Letters*, 1024–1027.

Yucek, T. & Arslan, H. (2009). A survey of spectrum sensing algorithms for cognitive radio applications. *IEEE communications surveys & tutorials*, 11(1), 116–130.

Zhang, Y., Xu, Y., Xu, Y., Yang, Y., Luo, Y., Wu, Q. & Liu, X. (2018). A multi-leader one-follower stackelberg game approach for cooperative anti-jamming: no pains, no gains. *IEEE Communications Letters*, 22(8), 1680–1683.