UNIVERSITÀ DEGLI STUDI
DI GENOVA

Queen Mary
University of London

Doctoral Thesis
Joint Doctorate in Interactive and Cognitive Environments

PHY-layer Security in Cognitive Radio Networks through Learning Deep Generative Models: an AI-based approach

---

by

Andrea Toma

A thesis submitted for the degree of
Doctor of Philosophy

Scuola Politecnica - Ingegneria

Dipartimento di Ingegneria Navale, Elettrica, Elettronica e delle Telecomunicazioni - DITEN

April 2020

PHY-layer Security in Cognitive Radio Networks through Learning Deep Generative Models: an AI-based approach

Andrea Toma

Joint Doctorate in Interactive and
Cognitive Environments
JD-ICE

XXXII cicle

Acknowledgements

This PhD Thesis has been developed in the framework of, and according to, the rules of the Joint Doctorate in Interactive and Cognitive Environments JD-ICE with the cooperation of the following Universities:

Università degli Studi di Genova (UNIGE)

DITEN - Dept. of Electrical, Electronic, Telecommunications Engineering and Naval Architecture

ISIP40 - Information and Signal Processing for Cognitive Telecommunications

https://isip40.it/

Supervisor: prof. Carlo Regazzoni
Co-supervisor: prof. Lucio Marcenaro

Queen Mary University of London (QMUL)

EECS - School of Electronic Engineering and Computer Science

CSI - Centre for Intelligent Sensing

http://cis.eecs.qmul.ac.uk/

Supervisor: prof. Yue Gao

April 2020

"*There is no such thing as a disembodied mind. The mind is implanted in the brain, and the brain is implanted in the body.*"

Antonio Damasio

Thanks to my supervisors Professor Carlo Regazzoni, Professor Yue Gao, and Professor Lucio Marcenaro I have successfully finalized my PhD, and this has been possible thanks to their guidance and work. With their experience and patient, each of them has taught me to autonomously conduct my research toward the proposed objectives. Equally, I want to thank Professor Andrea Cavallaro who is part of the research team, my independent assessor Professor Mirco Raffetto and the two referees Professors Giorgio Matteo Vitetta and Claudio Sacchi.

I am really grateful to have had this opportunity to be part of the joint doctorate under the JD-ICE programme with the collaboration between University of Genoa and Queen Mary University of London. In this regard, I would like to express my sincere gratitude to Mrs Melissa Yeo and Dr Riccardo Mazzon, the two research student coordinators who constantly assisted me and their valuable work allowed me to successfully manage my mobility and experience abroad.

Special thanks to Tassadaq Nawaz since his guidance helped me during my first year when I was not familiar with the topic; to Mohamad Baydoun, Rana Massoud, and Humaira Abdul Salam: the other JD-ICE candidates of the XXXII cycle; to Ali Krayani and Muhammad Farrukh with whom I shared most of my PhD period and collaborated for the same topic toward a common objective; and to Haoran Qi who spent part of his precious time to guide me when I was in the antenna group.

I will never forget all the other UniGe and QMUL PhD students, too many to list their names in these few lines. Some of them became friends of mine, but all of them made my PhD a unique and unrepeatable experience for me.

However, it was not only university and research experience with hard work. Indeed, these three years also allowed me to discover many different cultures and languages, as well as food and habits from other countries. I also met almost by chance, and for this reason I felt to be lucky, two of my best friends that I would like to mention here: He Wang, a PhD student in paleontology with whom I visited part of London even under the rain, and Junko Nuka, an engineer from Japan who taught me how to be more confident. Finally, a few words to Nisrine who was also there with her kindness and sincerity that have accompanied me during the conclusive part of my PhD, till writing-up and submission of this thesis.

I want to thank my former research team, in particular Professors Giuseppe Ricci, Francesco Bandiera, Angelo Coluccia (from the University of Salento where I graduated with a master's degree), and Professor Fabio Ricciato.

Conclusive thought is obviously for my parents Giuseppe and Maria, my sister Tiziana, my brother Pierluigi, my sister-in-law Luisa, my niece Morgana, my nephew Diego who are part of me, and all my friends from the warm southern part of Apulia.

To my family

Abstract

Recently, Cognitive Radio (CR) has been intended as an intelligent radio endowed with cognition which can be developed by implementing Artificial Intelligence (AI) techniques. Specifically, data-driven Self-Awareness (SA) functionalities, such as detection of spectrum abnormalities, can be effectively implemented as shown by the proposed research. One important application is PHY-layer security since it is essential to establish secure wireless communications against external jamming attacks.

In this framework, signals are non-stationary and features from such kind of dynamic spectrum, with multiple high sampling rate signals, are then extracted through the Stockwell Transform (ST) with dual-resolution which has been proposed and validated in this work as part of spectrum sensing techniques.

Afterwards, analysis of the state-of-the-art about learning dynamic models from observed features describes theoretical aspects of Machine Learning (ML). In particular, following the recent advances of ML, learning deep generative models with several layers of non-linear processing has been selected as AI method for the proposed spectrum abnormality detection in CR for a brain-inspired, data-driven SA.

In the proposed approach, the features extracted from the ST representation of the wideband spectrum are organized in a high-dimensional generalized state vector and, then, a generative model is learned and employed to detect any deviation from normal situations in the analysed spectrum (abnormal signals or behaviours). Specifically, conditional GAN (C-GAN), auxiliary classifier GAN (AC-GAN), and deep VAE have been considered as deep generative models.

A dataset of a dynamic spectrum with multi-OFDM signals has been generated by using the National Instruments mm-Wave Transceiver which operates at 28 GHz (central carrier frequency) with 800 MHz frequency range. Training of the deep generative model is performed on the generalized state vector representing the mmWave spectrum with normality pattern without any malicious activity. Testing is based on new and independent data samples corresponding to abnormality pattern where the moving signal follows a different behaviour which has not been observed during training.

An abnormality indicator is measured and used for the binary classification (normality hypothesis otherwise abnormality hypothesis), while the performance of the generative models is evaluated and compared through ROC curves and accuracy metrics.

Contents

Abstract **xi**

1 Introduction to this thesis **1**
1.1 Motivation and Objectives 1
1.2 Thesis Contributions 2
1.3 Thesis Organization 3
1.4 List of publications and under review work 6
Bibliography 7

2 The CR and PHY-layer security with real applications **9**
2.1 An overview on cognitive radio and physical layer security against jamming attacks 9
2.2 Cognitive radio applications and research 11
2.2.1 The SHIELD project and Smart SPD-Driven Transmission layer 11
2.2.2 TVWS technology 12
2.2.3 5G and the mmWave technology 13
2.3 Software defined radio systems for CR 14
2.3.1 The Selex testbed 14
2.3.2 The TVWS hardware 17
2.3.3 The NI millimeter-wave system 19
Bibliography 21

3 Spectrum representation and feature extraction **25**
3.1 Spectrum sensing and corresponding methods in the literature . . . 25
3.2 Conventional features and detectors 26
3.3 Example of feature extraction: spectrum correlation in modulated signals 28
3.3.1 Data acquisition and processing 29
3.3.2 Cyclostationary feature analysis 33
3.3.3 α-profile extracted from signals with different modulation schemes 38

3.4 Non-stationarity and time-frequency analysis . . . 39
3.5 High sampling rate dynamic signals and ST . . . 43
3.5.1 Problem formulation . . . 44
3.5.2 Discrete ST and the dual-resolution method . . . 44
3.5.3 Data Acquisition of real dynamic signals . . . 47
3.5.4 Validation and comparison with STFT . . . 48
3.5.5 Conclusion and Future Directions . . . 54
Bibliography . . . 56

4 Learning Dynamic Models from spectrum representation 63
4.1 Motivation . . . 63
4.2 Example (cont'd): Spectrum correlation in modulated signals . . . 65
4.2.1 Cyclic Spectrum Intelligence (CSI) algorithm . . . 66
4.2.2 Learning and acting algorithms . . . 69
4.2.3 Validation of the proposed algorithm . . . 70
4.3 Probabilistic dynamic signal representation . . . 76
4.3.1 Single entity state . . . 77
4.3.2 Interacting entities situation assessment . . . 81
4.4 Learning dynamic Bayesian representations . . . 83
4.4.1 Learning vocabulary, state and state changes . . . 83
4.4.2 Learning causal conditioned distributions . . . 85
4.4.3 Learning interactions . . . 87
4.4.4 Some current learning techniques and probable future directions . . . 88
4.5 Application directions of ML techniques to cognitive dynamic jamming 89
4.6 Conclusion and future directions . . . 92
Bibliography . . . 93

5 AI principles and Deep Generative Models 97
5.1 AI basis: self-aware systems and the free-energy principle . . . 97
5.2 AI and Self-Awareness in CR . . . 99
5.3 Deep learning for data-driven SA . . . 101
5.4 Proposed SA functionality: spectrum abnormality detection . . . 103
5.5 Deep generative models for spectrum abnormality detection . . . 104
5.6 GAN, C-GAN, AC-GAN, and VAE . . . 105
Bibliography . . . 111

6 AI-based spectrum abnormality detection: system model 117
6.1 Related work . . . 117

6.2 Approach based on Dynamic Bayesian Networks 119
6.3 The proposed approach based on Deep Generative Model 120
6.3.1 Feature extraction through ST and generalized state vector . 121
6.3.2 Deep Generative Model Block: C-GAN/AC-GAN 122
6.3.3 Deep Generative Model Block: VAE 128
6.3.4 Abnormality detection and performance evaluation 131
Bibliography 132

7 AI-based spectrum abnormality detection: experimental results 135
7.1 Introduction 135
7.2 Training of the C-GAN model 136
7.3 C-GAN as abnormal behaviour detector 138
7.4 Comparison with other methods 143
7.4.1 Comparison with conventional methods 143
7.4.2 Comparison with AI-based methods 144
7.5 Abnormality detection through AC-GAN and VAE 144
7.6 Conclusion and future work 147
Bibliography 148

8 Conclusion of this thesis and future work 151

1 Introduction to this thesis

1.1 Motivation and Objectives

The proliferation of wireless devices due to the emerging technologies has elevated spectrum scarcity problem. There has been a risk that spectrum might be congested and more users cannot be facilitated. However, measurements have shown that a large portion of the spectrum experiences low utilization. To cope with such a problem, Cognitive Radio (CR) is envisioned as a potential candidate that enhances spectrum efficiency and abstains network from spectrum under-utilization issue [8]. CR has been defined to exhibit three integral attributes which are *observations*, *reconfiguration*, and *cognition* (a hierarchical cognition cycle is described in [10]). In the observation process, CR gathers information about the radio environment. In the reconfiguration step, radio parameters are adjusted or changed. Whereas, cognition is related to understanding the radio environment, taking decisions on gathered information and learning the implications of such decisions on radio performance [2]. Learning and reasoning are fundamental aspects of cognition that may be achieved if the CR network subsumes a certain degree of *Self-Awareness (SA)* which can be developed by implementing Artificial Intelligence (AI) techniques. SA has been defined in the literature as the ability of a system to generate knowledge about itself and its environment and determine actions to be executed based on that knowledge [1, 3, 4]. A basic data-driven SA module should include the following functionalities: *i)* the ability of a CR device to autonomously learn *Generative Models* by observing its states and the occurring environmental changes simultaneously; *ii)* the ability of a CR device to decide whether communications inside the radio spectrum occur according to a normal behaviour among the device itself and other devices (*Abnormality Detection*); *iii)* the detected abnormalities can be used either by the control system to apply *Abnormality Mitigation* strategies or by the SA module itself to *Incrementally Learn* new models that describe different dynamic situations not included in previous experiences. This can be performed by minimizing the free energy, i.e., minimizing the difference between the system's state and observation [11]. The free energy is a function that represents the prediction error between sensory input and a given generative model, namely a model that can generate sensory samples and their causes. Theoretically, the free energy principle illustrates how autonomous systems minimize

the variations of free energy.

Introducing a SA module in CR can support the system to improve the decision and action cycles after detecting abnormal behaviours such as jammer attacks that can manipulate the radio spectrum and teach the CR malicious behaviours. Thus, the SA module can lead to establishing secure networks against various attacks.

1.2 Thesis Contributions

This work addresses on the first two functionalities of SA, described in Sec. 1.1, which are essential to achieve the third functionality. Specifically, an AI-based abnormality detection approach, based on learning deep generative models, is proposed to enhance the physical (PHY)-layer security in the CR by detecting abnormal behaviours inside the radio spectrum. Generative models are dynamic models used for probabilistic reasoning on the observed data [6]. In this work, the observed data is clustered from a generalized state vector defined in ref. [5]. Then, an adaptive prediction is performed depending on the current generalized state which consists of the current state and its first-order derivative. In particular, the Conditional Generative Adversarial Network (C-GAN), the Auxiliary Classifier Generative Adversarial Network (AC-GAN), and the Variational Auto-Encoder (VAE) are employed as learning block and their performance is evaluated and compared. GAN-based models are the most crucial research avenues in the field of AI as supervised and unsupervised learning techniques and their outstanding data generation capacity has received extensive attention [9, 7]. Consequently, a learning block based on GAN is proposed in this work due to its fast and accurate inferences which are based on a likelihood-free algorithm. Nevertheless, GAN tends to lack full support over the data. For this reason, VAE is also investigated as learning block because, like other likelihood-based models, it is better density model in terms of the likelihood criterion, but it generates more dispersed samples.

However, the proposed models are characterized by the data dimensionality and the PHY-layer level at which the AI-method is implemented. Indeed, the abnormality detection is applied just after the receiving antenna and the down-conversion process (before the demodulation block) where high dimensionality data, which represents the multi-signals wideband spectrum with high sampling rate, is extracted.

Through a probabilistic model like Dynamic Bayesian Network (DBN), it is possible to learn switching models from data series of generalized states where each switching variable can be associated with a different linear dynamic model. But, in the case of high dimensionality data, this would generate a vast vocabulary of switching variables, making the model computationally intractable. For this reason, DBN can

be employed in applications where the number of possible switching dynamic models to be included in different Bayesian filters is limited (low dimensionality data).

On the contrary, GAN- and VAE- based methods can effectively manage a high number of different dynamic models implicitly. As the main drawback, they are unable to manage uncertainty as DBN does with probabilistic knowledge.

To sum up, the main contributions of this work are as follows:

- Abnormality detection in the PHY-layer of CR through C-GAN, AC-GAN, or VAE.
- The potential of the proposed method lies in a fact that it can be incorporated in a CR system where learning directly from the spectrum which generates high dimensional data is required.
- The proposed method follows a data-driven approach where no explicit mathematical models are needed, and unknown probabilistic models are automatically learned from data even with high complexity and without any prior knowledge about the signal.
- The proposed method helps in developing the SA module in CR by learning generative models and detecting abnormalities. This allows achieving the third functionality of SA defined previously.

Theoretical aspects of the techniques and methods employed in this thesis are introduced along with analysis of the state-of-the-art to provide a general framework about related work and current directions for the topic of this thesis. Each part of the proposed approach for spectrum abnormality detection through AI, from spectrum representation to detection of abnormal situations in the spectrum, is then described in details throughout the thesis. Experiments have been conducted on real data collected during the PhD research activity by using existing testbeds for CR with SDR components. Results obtained by applying the real datasets to the implemented algorithms are then provided along with comparisons with previous results from different approaches. Some possible future work is also mentioned throughout the thesis. The overall organization of the thesis is further clarified in the following section.

1.3 Thesis Organization

The general idea of the research presented throughout this thesis is represented by the diagram of the cognitive capability for spectrum abnormality detection in Fig. 1.1.

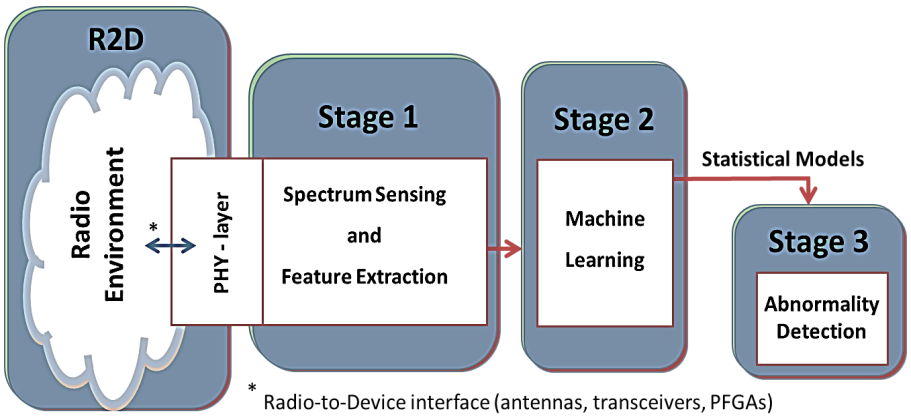


FIGURE 1.1: General diagram of the cognitive capability as part of the work investigated in this thesis

In this section, each block in the diagram is introduced with a short description which also points to the corresponding chapters within the thesis.

- **Radio Environment and the PHY-layer:** this constitutes the **Radio-to-Device (R2D) interface**, i.e., the interface by means of which radio devices can access the shared spectrum. R2D enables dynamic and opportunistic spectrum access which is the main functionality in CR devices. A detailed description can be found in **Chapter 2** where CR and the PHY-layer security are introduced with some real applications and hardware including the SHIELD project and the Smart SPD-Driven Transmission layer with the Selex testbed, the TVWS technology with a Broadband Radio transmitter and a RFeye Node receiver system, and the 5G and mmWave technology with the NI millimeter-wave system.

- **Spectrum Sensing and Feature Extraction:** this is the **stage 1** of the cognitive capability and presented in detail in **Chapter 3** which, firstly, introduces the context of spectrum sensing from the literature and some conventional features and detectors. As a practical example, feature extraction is shown through a spectrum correlation-based method in modulated signals; this method has been exploited in a cycle spectrum intelligence algorithm investigated as part of the PhD activity. Data acquisition is performed on the Selex dataset. In this case, the α-profile of real modulated signals is extracted and used as feature.

  Afterwards, a time-frequency method for non-stationary signals is discussed; this is employed as spectrum representation method in the proposed work in Fig. 1.1 by which features related to time and frequency can be extracted. To this end, an approach for high sampling rate signals based on the ST has been

proposed and validated on the Selex dataset. Results and a comparison with the STFT are then presented.

- **Machine Learning:** this block has been implemented to enable a data-driven self-awareness. This **stage 2** is divided up into two chapters of this thesis. In **Chapter 4**, the cycle spectrum intelligence algorithm is further analysed as an application of learning an unkwown model from the extracted feature, namely the α-profile. In this example, the learned neural network acts as a binary classifier. After discussing the results and comparison with previous works, this chapter goes through a thorough analysis of the state-of-the-art on learning dynamic models, from probabilistic dynamic signal representation to learning dynamic Bayesian representations. An application of ML techniques to cognitive dynamic jamming is then presented.

  While, **Chapter 5** goes beyond machine learning by introducing recent advancements through two AI concepts, self-awareness and the free-energy principle, which are fundamental to build self-aware devices. This approach can also be implemented in CR. Indeed, as part of this thesis, deep generative models have been implemented for an effective data-driven SA in CR. Spectrum abnormality detection is described along with C-GAN, AC-GAN, and VAE models.

- **Abnormality Detection:** after a brief discussion about related work and the approach based on DBN, the proposed approach, represented by a deep model-based Abnormality Detection scheme for CR, is discussed in **Chapter 6** corresponding to **Stage 3** shown in Fig. 1.1. In this chapter, a detailed algorithm description can be found along with details of the implementation for each of the deep generative models considered in Chapter 5. The performance metrics, that have been used to evaluate these models, are also described.

  The experimental results of the AI-based spectrum abnormality detection algorithm are then discussed in **Chapter 7**. Data acquisition is performed on the mmWave dataset, collected by using the NI system, on which ST analysis is applied to extract time-frequency related features. The performance of the three generative models, used as abnormality detectors, are presented in details.

Chapter 8 provides an overview about Conclusion and Future work related to this thesis.

To sum up, the thesis structure is represented in Tab. 1.1.

	Chapter
	1. *Introduction to this thesis*
R2D	2. *The CR and PHY-layer security with real applications*
Stage 1	3. *Spectrum representation and feature extraction*[1,3]
Stage 2	4. *Learning Dynamic Models from spectrum representation*[2,3] 5. *AI principles and Deep Generative Models*
Stage 3	6. *AI-based spectrum abnormality detection: system model* 7. *AI-based spectrum abnormality detection: experimental results*[4,5]
	8. *Conclusion of this thesis and future work*

TABLE 1.1: Thesis structure

1.4 List of publications and under review work

- **Accepted publications:**

[1] [conference paper] Toma A., Nawaz T., Marcenaro L., Regazzoni C., Gao Y., "Exploiting ST-Based Representation for High Sampling Rate Dynamic Signals". In: *Woungang I., Dhurandher S. (eds) 2nd International Conference on Wireless Intelligent and Distributed Environment for Communication. WIDECOM2019*. Lecture Notes on Data Engineering and Communications Technologies, vol. 27, Springer, Cham, pp. 203-217, DOI https://doi.org/10.1007/978-3-030-11437-4_16, 28 March 2019 (Workshop best paper)

[2] [book chapter] Toma A., Regazzoni C., Marcenaro L., Gao Y., "Learning Dynamic Jamming Models in Cognitive Radios", in *Cognitive Radio Applications and Practices, Handbook of Cognitive Radio, Springer, 2018*. doi:10.1007/978-981-10-1389-8_64-1, June 2018

[3] [journal paper] A. Toma, T. Nawaz, Y. Gao, L. Marcenaro and C. S. Regazzoni, "Interference mitigation in wideband radios using spectrum correlation and neural network," in *IET Communications*, vol. 13, no. 10, pp. 1336-1347, doi: dx.doi.org/10.1049/iet-com.2018.5720, 25 June 2019

[4] [journal paper] Toma A., Krayani A., Farrukh M., Qi H., Marcenaro L., Regazzoni C., Gao Y., "AI-based Abnormality Detection at the PHY-layer of Cognitive Radio by Learning Generative Models", in *IEEE Transactions on Cognitive Communications and Networking (TCCN)*, special issue *Evolution of Cognitive Radio to AI Radio and Networks*, doi: 10.1109/TCCN.2020.2970693, 30 Jan 2020

- **Under review work:**

[5] [conference paper] Toma A., Krayani A., Marcenaro L., Gao Y., Regazzoni C., "Deep Learning for Spectrum Anomaly Detection in Cognitive mmWave Radios", submitted to *European Wireless (EW) 2020* conference, Verona, Italy, May 6-8, 2020

Bibliography

[1] R. Andrade and J. Torres. Self-awareness as an enabler of cognitive security. In *2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, pages 701–708, Nov 2018.

[2] D. D. Clark, C. Partridge, J. C. Ramming, and J. T. Wroclawski. A knowledge plane for the internet. In *Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, SIGCOMM '03, pages 3–10, New York, NY, USA, 2003. ACM.

[3] A. Diaconescu, B. Porter, R. Rodrigues, and E. Pournaras. Hierarchical self-awareness and authority for scalable self-integrating systems. In *2018 IEEE 3rd International Workshops on Foundations and Applications of Self-Systems (FAS-W)*, pages 168–175, Sep. 2018.

[4] SDR Forum. Cognitive radio definitions and nomenclature. *[online], available: https://www.wirelessinnovation.org/*, 01 2008.

[5] K. Friston, B. Sengupta, and G. Auletta. Cognitive dynamics: From attractors to active inference. *Proceedings of the IEEE*, 102(4):427–445, April 2014.

[6] X. Gao, Z. Y. Zhang, and L. M. Duan. A quantum machine learning algorithm based on generative models. *Science Advances*, 4(12), 2018.

[7] A. Grover, M. Dhar, and S. Ermon. Flow-gan: Bridging implicit and prescribed learning in generative models. *CoRR*, abs/1705.08868, 2017.

[8] S. Haykin. Cognitive radio: brain-empowered wireless communications. *IEEE Journal on Selected Areas in Communications*, 23(2):201–220, Feb 2005.

[9] Z. Pan, W. Yu, X. Yi, A. Khan, F. Yuan, and Y. Zheng. Recent progress on generative adversarial networks (gans): A survey. *IEEE Access*, 7:36322–36333, 2019.

[10] J. Wang, S. Feng, Q. Wu, X. Zheng, and Y. Xu. Hierarchical cognition cycle for cognitive radio networks. *China Communications*, 12(1):108–121, Jan 2015.

[11] H. Zaal, H. Iqbal, D. Campo, L. Marcenaro, and C. Regazzoni. Incremental learning of abnormalities in autonomous systems. In *16-th IEEE International Conference on Advanced Video and Signal-based Surveillance (AVSS), Taipei, Taiwan*, 10 2019.

2 The CR and PHY-layer security with real applications

The **Radio-to-Device (R2D)** interface in the general cognitive capability diagram of a wireless device consists of hardware components at the physical (PHY)-layer level such as antennas, analog/digital transceivers, and FPGAs (see Fig. 2.1). This kind of interface connects the wireless device to the radio environment where the signals are transmitted in a shared spectrum. In the following, the cognitive radio technology is introduced through presenting the latest applications such as physical layer security which is fundamental in R2D communications due to the broadcast nature of wireless communications. Finally, three real software defined radio (SDR) systems are introduced.

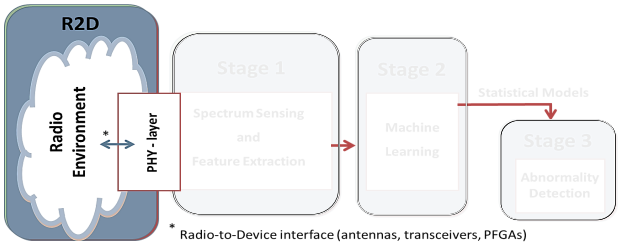


FIGURE 2.1: Radio-to-Device (R2D) interface

2.1 An overview on cognitive radio and physical layer security against jamming attacks

The explosive rise in the amount of wireless equipment, including Internet of Things (IoT) and Vehicle to Everything (V2X) devices will support tremendous wireless connectivity causing the spectrum scarcity [17, 30]. Cognitive Radio (CR) has then been proposed to address such an issue and increase the radio spectrum utilization [25]. CR allows secondary users to sense frequently and access opportunistically the spectrum bands which are not in use by primary licensed users and without affecting the quality of service [29, 13, 11, 16].

Besides, fifth-generation (5G) technology will provide a system structure for these emerging V2X and IoT applications that require high reliability for secure message delivery between transmitters and receivers; this imposes the need of an efficient hybrid access scheme for licensed and unlicensed spectrum. CR can manage such a dynamic spectrum access. A cognitive radio network is shown in Fig. 2.2.

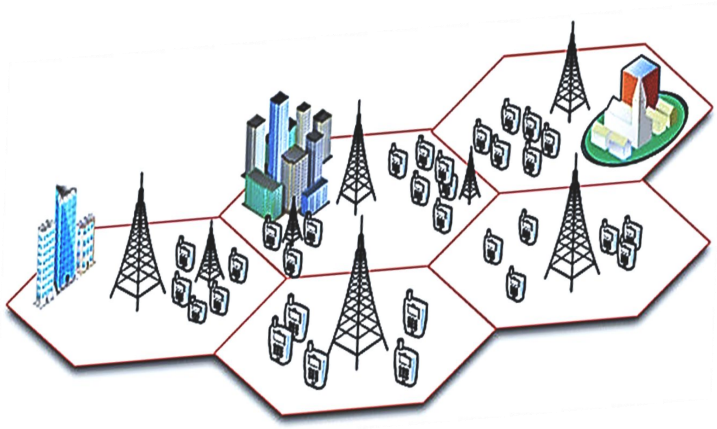


FIGURE 2.2: An example of cognitive radio network [2]

In this context, physical (PHY-)layer security in CR has attracted broad interest to achieve secured communications that involve multiple signal transmissions due to the shared wideband spectrum and the coexistence in tight integration with different wireless systems [32, 27]. Such an open access medium and dynamic environment makes the system vulnerable to malicious users, like jammers [14, 34], that aim to manipulate the radio spectrum by injecting anomalous signals and enforce the system to learn wrong behaviours that lead the radio to take mistaken actions [26, 28]. Therefore, stringent security requirements have been established and precise detection of spectrum anomalies is fundamental to enhance the physical layer security and improve the system's performance which is the main objective of the work in this thesis. To this end, autonomous learning is a crucial component in CR system to adapt to the perceived wireless environment and potentially maximize the utility of the available spectrum resources and allow the radio to take optimal decision and act efficiently [33].

Both legitimate users and jammers can quickly learn the transmission parameters of other users in the spectrum of interest and adaptively adjust its transmission parameters to maximize the utility function and the damaging effect, respectively [31]. Anti-jamming methodologies include retroactive frequency hopping (namely channel surfing), transmission power changing, modifying the modulation of the transmitted waveform, as well as spread spectrum techniques. In particular, when channel surfing is applied, the operating frequency of the radio is changed whenever a strong interfering signal is detected. The probability of detection and reconstruction accuracy of a signal may be enhanced by increasing the transmission power and/or gain. Modulation altering may also be decisive in alleviating RF jamming interference, provided that the radios are equipped with automatic modulation recognition capabilities, allowing them to detect and classify the modulation-related features of jamming

waveforms. Spread spectrum techniques use data-independent, random sequences to spread a narrowband information signal over a wide (radio) band of frequencies. Under the premise that it is hard or infeasible for an attacker to jam the entire frequency band, the receiver can correlate the received signal with a replicate of the random sequence to retrieve the original information signal and, in this way, any interference is rejected.

2.2 Cognitive radio applications and research

Research in CR and PHY-layer security has been carried on in different fields for both civil and military purposes. In this section, three current applications are introduced: the *SHIELD project* along with *SPD-Driven Smart Transmission layer* which implement Spectrum Intelligence for security against jamming attacks; *TVWS* to mitigate the shortage of wireless bands; and millimeter-wave communications for future 5G technology.

2.2.1 The SHIELD project and Smart SPD-Driven Transmission layer

Nowadays, cyber-physical systems (CPSs) and Internet of Things (IoT) are rapidly expanding [8] and new business opportunities are being developed thanks to the dynamic interaction between the entities involved in the business.

The need for measurable security in the context of interoperating services, applications, systems, and devices in a Cyber-Physical-Systems (CPSs) and Internet-of-Things (IoT) [9] framework requires the development of an appropriate paradigm. A step forward in that direction is made by *SHIELD* which consists of methodologies for building secure embedded systems [8]. Specifically, the basic approach specifies security through the terms Security (S), Privacy (P) and Dependability (D).

Dynamic interactions between entities represent the recent evolution of collaborations between entities for Internet-based services. In this context, autonomous decisions are enabled by dynamic modelling but the lack of a measurable security makes information exchanges one of the big challenges. Current research on security in CPSs is far less intensive than research on security in computing and networking leaving many devices vulnerable to attacks.

To address the problem of jamming attacks, Fig. 2.3, and provide safe and reliable communication in wireless environments, according to the minimum requirements for smart and secure data transmission in shared spectrum, SDR-capable devices are

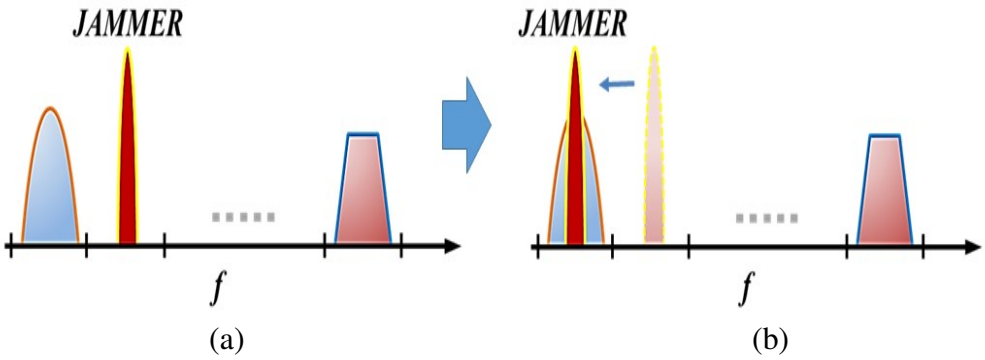


FIGURE 2.3: Graphical representation of a jamming attack in a wireless spectrum: (a) before the attack and (b) the jammer is in the same band of the normal transmission

based on a set of services implemented at the network level to form a *smart SPD-driven transmission layer* [8]. This is made possible thanks to both the SDR technology with its reconfigurability properties and the CR technology with its learning and self-adaptive capabilities. Cognitive functionalities include self-awareness, spectrum awareness, spectrum intelligence, jamming detection and counteraction, and self-protection.

2.2.2 TVWS technology

In recent years, the rapid development of wireless devices and wireless services resulted in a ever growing demand and shortage of the wireless spectrum [22]. For this reason, CR was proposed to efficiently utilize spectrum resources by allowing unlicensed usage of vacant spectrum. Indeed, as reported by the Federal Communications Commission (FCC) and the UK Office of Communications (Ofcom), a large percentage of spectrum resources is underutilized. This has encouraged the governments to take critical steps towards releasing multiple bands for dynamic spectrum sharing. Consequently, spectrum holes in the licensed spectrum can now be used by secondary users (SUs) without causing any interference to primary users (PUs). In particular, TV white spaces (TVWS) [1, 10, 21, 24, 19] is one of the most promising paradigm for dynamic spectrum sharing in the digital TV band; this allows secondary users to use licensed spectrum bands provided that they change their access strategies to avoid or reduce interference to the primary users. Dynamic Spectrum Access (DSA), or dynamic channel access, assumes an important role in TVWS cognitive radio networks [20].

To this end, SS is a promising solution to identify potential spectral holes and is

one of the most challenging tasks in CR networks [23]. Cooperative SS (CSS) is an effective approach to achieve significant performance gain in detecting spectrum holes, by exploiting the spatial diversity of collaborative secondary users (SUs). However, due to the openness of low-layer protocol stacks, CSS networks are vulnerable to attacks from SS data falsification (SSDF). The main goals of malicious attacks come from two aspects: decreasing detection probability for disturbing the normal operation of PUs and increasing false alarm probability to deprive access opportunities for honest SUs. In decentralized CSS networks, sensing results are exchanged between neighboring SUs to improve the network reliability to link failure. However, this characteristic makes decentralized CSS more vulnerable to malicious attacks, as observations at honest SUs are also available to malicious users during the information exchanging and convergence process. Furthermore, corrupted data can be integrated into the decisions of honest neighbour SUs, which eventually brings significant performance degradation to the whole CSS network.

Recently, the problem of malicious user detection and jamming attacks have also been addressed by TVWS research in the digital TV band in [23].

2.2.3 5G and the mmWave technology

Recently, substantial efforts have also been devoted to the research and development of the fifth-generation (5G) mobile systems. Specifically, the significant advantages offered by the propagation characteristics in terms of frequency re-usability and large channel bandwidths make millimeter-wave (mmWave) suitable for the very high capacities required by the fifth generation (5G) wireless communication system [18, 7]. Indeed, mmWave can achieve Giga-bits/sec data rate and large-data-capacity. Consequently, high interest in this part of the electromagnetic spectrum has risen in the recent years and on October 22nd 2015, the Federal Communications Commission (FCC) proposed new rules (FCC 15138) for wireless broadband frequencies of 28 GHz, 37 GHz, 39 GHz and 64 - 71 GHz bands. The 71-76 GHz and 81-86 GHz bands are also being considered as a potential candidate for mmWave mobile services in the USA in the context of the Commission's spectrum frontiers proceeding (July 2016, FCC 16-89 - GN Docket No. 14-177). The very small wavelengths of mmWave signals combined with advances in technology for miniaturized antennas forming multiple antenna systems provide very high gain and electrically steerable arrays. On the other hand, the development of technologies in the mmWave bands faces significant technical obstacles: 1) the increase in omnidirectional free space path loss with higher frequencies - this issue can be compensated by an increased antenna gain obtained through appropriate beamforming; and 2) mmWave signals

can be severely vulnerable to shadowing resulting in outages, rapidly varying channel conditions and intermittent connectivity - this issue is particularly concerning in cluttered, urban deployments where coverage frequently requires non-line-of-sight (NLOS) links. In ref. [3], measurements of mmWave outdoor cellular propagation at 28 and 73 GHz in New York City, NYC, USA, are used to derive statistical channel models. The urban canyon environment of the tested scenario lacked of line-of-sight (LOS) links. It has been found that, even in highly NLOS environments, strong signals can be detected 100–200 meters from potential cell sites. However, enormous amount of under-utilized bandwidth lies in the millimeter-wave bands; for example, the available spectrum at these frequencies can be easily 200 times greater than all cellular allocations. Again, DSA plays and important role in the recent 5G technology [15], but strict security requirements are desired for the 5G systems, since more and more sensitive information will be transmitted wirelessly. To this end, physical-layer security will be a beneficial complement to conventional security mechanisms [34].

In the following section, experimental hardware for each of the three applications, used to generate real data thanks to SDR devices, are described in detail.

2.3 Software defined radio systems for CR

2.3.1 The Selex testbed

In the framework of the SHIELD project and Smart SPD-Driven Transmission layer, the testbed is a SDR platform, Fig. 2.4, which consists of two Secure Wideband Multi-role - Single-Channel Handheld Radios, SWAVE HHs; the first one is the transmitter

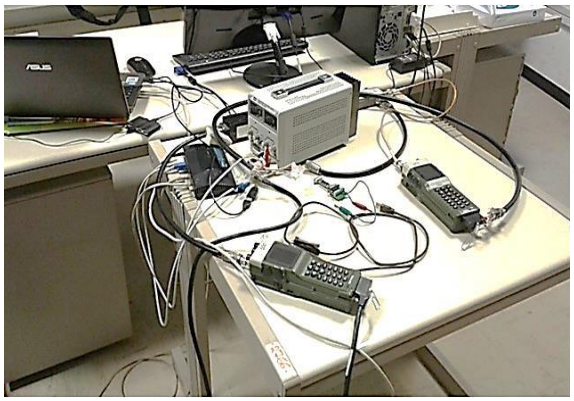


FIGURE 2.4: SDR testbed in the SHIELD project to generate wideband spectrum measurements: hardware platform

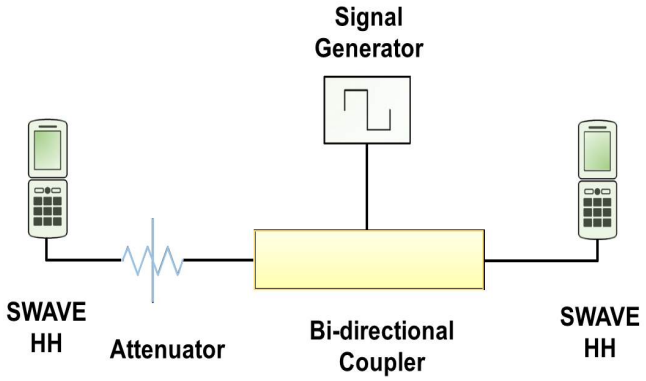


FIGURE 2.5: Diagram of the main components of the testbed and their connections

while the second one receives the wideband signal, connected through a dual directional coupler [6]. The general diagram of the testbed is shown in Fig. 2.5. The fully operational SDR radio terminal SelexES (2013), SWAVE HH (from now on referred to as HH), is capable of generating a multitude of wideband and narrowband waveforms. Currently, two functional waveforms are installed on the radio: 1) SelfNET Soldier Broadband Waveform (SBW), whose channel bandwidth is adjustable in the range 1.25 MHz to 5 MHz with channel spacing of 2 MHz and data is modulated using a fixed digital modulation; 2) VHF/UHF Line Of Sight (VULOS), which supports two analog modulation techniques, Amplitude Modulation (AM) and Frequency Modulation (FM), while both channel bandwidth and channel spacing are adjustable up to 25 kHz [5].

The radio provides operability in both VHF (30 - 88 MHz) and UHF (225 - 512 MHz) bands. When VHF is selected, analog to digital conversion is performed directly at RF and the frequency band scanned is always 0-120 MHz; while in the case of UHF, the conversion is performed at intermediate frequency (IF) and the frequency band scanned depends on the center carrier frequency f_c of the radio ($[f_c - 35; f_c + 85]$ MHz). No selective filtering is applied before analog-to-digital conversion. Broadband digitized signal is then issued to the field-programmable gate array (FPGA), where it undergoes digital down conversion, matched filtering and demodulation.

Several interfaces are provided by the hypertach expansion placed at the bottom of each HH, specifically, 10/100 Ethernet, USB 2.0, RS-485 serial, DC power interface (max 12.7V), and PTT. The software architecture of the radio is compliant with the Software Communications Architecture (SCA) 2.2.2 standard.

Maximum transmission power of the HH is 5W, with the harmonics suppression at the transmitter side over -50 dBc. Superheterodyne receiver has specified image rejection better than -58 dBc. Because of the high output power of the radios, one programmable attenuator is included in the communication path and programmed to

their maximum attenuation value -30 dB.

Agilent 778D 100MHz-2GHz dual directional coupler with 20dB nominal coupling is used as communication medium between the two HHs.

Guided propagation exhibits several important advantages with respect to the over-the-air implementation: accurate and stable RF levels, repeatability of the experiments without the uncertainties characteristic to wireless transmission, possibility to connect test instruments and generators, avoiding regulatory issues related to transmitting outside of the allowed frequency bands.

The testbed provides support for remote control of HH's transmission and reception parameters through Ethernet and the Simple Network Management Protocol (SNMP v3). Full details on the testbed architecture may be found in [5].

In the spectrum sensing process, the HH's 14-bit Analog-to-Digital-Converter (ADC) performs sampling at 250 Msamples/s. Every 3 seconds, a burst of 8192 consecutive samples is buffered, and then output over the serial port at 115200 bauds. The samples are then parsed and transformed into the frequency domain using the FFT as in Fig. 2.6. The bandwidth of the corresponding spectrum is 120 MHz wide around the center carrier frequency of the radio. Consequently, the effective resolution is 29.3 kHz/sample. In order to obtain higher frequency resolutions, two possible changes to the testbed are increasing the buffer size on the HHs, and finding ways to transfer spectrum data at higher baud rate. Additional equipment can be connected to the dual directional coupler such as a signal generator to inject further modulated signals into the spectrum of interest. Further details can be found in [5].

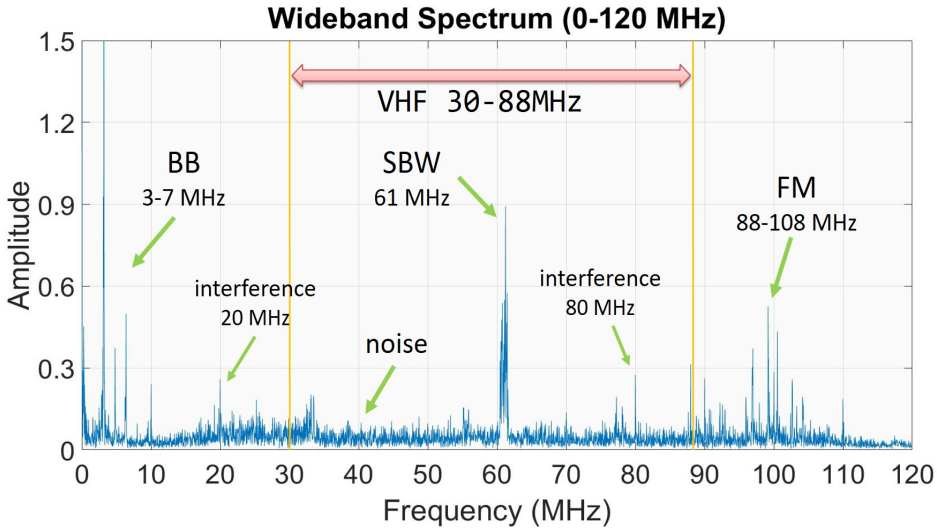
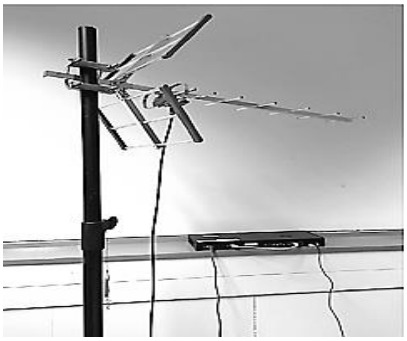


Figure 2.6: Spectrum observation in the 0-120 MHz (1 burst) which includes the SBW signal at 61 MHz and with transmit power equal to -3dBm



(a)

(b)

FIGURE 2.7: (a) Carlson RuralConnect® TX: Base Station; (b) Carlson RuralConnect® TX: Client

(a) Node with outdoor kit: front view

(b) Node with outdoor kit: rear view

(c) Indoor system

FIGURE 2.8: CRFS RFeye Node: receiver

2.3.2 The TVWS hardware

The testbed in the TVWS band (470-790 MHz, digital TV in the UK), shown in Figs. 2.7 and 2.8, consists of a receiver and a transmitter in the considered band as described below.

i. Carlson RuralConnect® TV White Space Broadband Radio transmitter (Fig. 2.7): it utilizes "white space" spectrum in the 470–608 MHz (TV channels 14–36) and 614–698 MHz (TV channels 38–51) frequencies. It is a fully compliant system with FCC Part 15H regulations in the UK. A base station (BS) is connected to the internet via a standard Ethernet port (see Fig. 2.7(a)). It encodes data from the Ethernet connection for transmission to one or more client stations (CPEs) units, Fig. 2.7(b), and decodes transmissions from the CPEs and sends these to the internet. Signals are transmitted from a directional Carlson Wireless Sector Antenna whose gain is 11dBi and its feeding cable has 1dBi loss, and the EIRP is 33dBm. The CPE is installed at the subscriber's premises and uses external antennas that operate over the entire band from 470 MHz to 698 MHz. CPEs connect to a BS via a UHF radio. A Log Periodic Directional UHF Antenna is connected to the client.

ii. RFeye Node 20-6 receiver (Fig. 2.8): benchmark for cost-effective, real-time 24/7 monitoring of the radio spectrum. It is provided by CRFS and based on hardware system either fix or mobile. Key features also include wide frequency range i.e. 10 MHz to 6 GHz, secure network connectivity using Secure Shell (SSH), high precision GPS, signal geolocation and correlation of signals. An Ultra-Wideband omnidirectional Cobham antenna in the frequency band 0.80 - 6.00 GHz is employed as a vertically polarised broadband antenna with 1-3dBi gain across the band. The data collected from the network can be used by various RFeye Software such as RFeye Site for real-time spectrum monitoring. Specifically, Node Communication Protocol (NCP) acts as an interface between any RFeye Node based hardware system and a client software application. RFeye NCP Toolbox is a MATLAB® software comprising of a library to create an NCP connection for interaction with the RFeye Node.

iii. The whole spectrum of interest (470-790 MHz) consists of 40 channels whose bandwidth is 8 MHz for each channel.

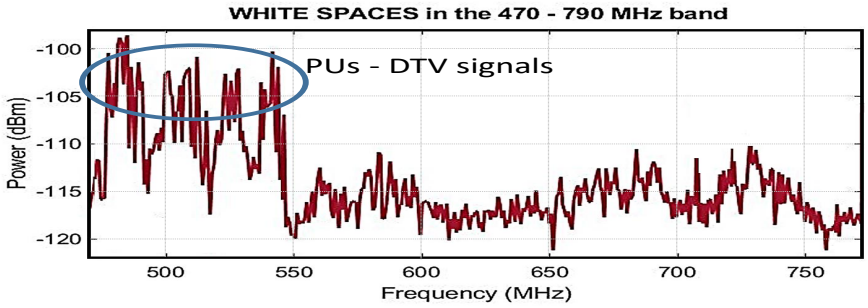


FIGURE 2.9: TVWS spectrum in the baseband with PUs

The corresponding RF spectrum is shown in Fig. 2.9 in which there are I = 40 channels over TVWS spectrum (each of them with bandwidth of 8 MHz) ranging from 470 MHz to 790 MHz.

2.3.3 The NI millimeter-wave system

The National Instruments mmWave Transceiver System at Queen Mary University of London in Fig. 2.10, used to collect the dataset with the objective of validating the proposed approach, is a Software Defined Radio (SDR) platform consisting of hardware equipment and application software that enables real-time over the air mmWave communications research. As shown in the diagram of Fig. 2.11, the transceiver system is comprised of chassis, controllers, a clock distribution module, 192 MS/s Field-Programmable Gate Array (FPGA) modules, high speed Digital-to-Analog Converters (DACs) and Analog-to-Digital Converters (ADCs) (3.072 GS/s), Local Oscillator (LO) and Intermediate Frequency (IF) modules, and mmWave radio heads (24.25 - 33.4 GHz) for upconversion from 12 GHz IF to mmWave band and downconversion from mmWave band to 12 GHz IF. A detailed description can be found in [12].

The radio heads are connected to a Ka-band circular horn transmitting antenna (26-40 GHz) and a slot antenna at 28.5 GHz (which can be seen in the lower part of

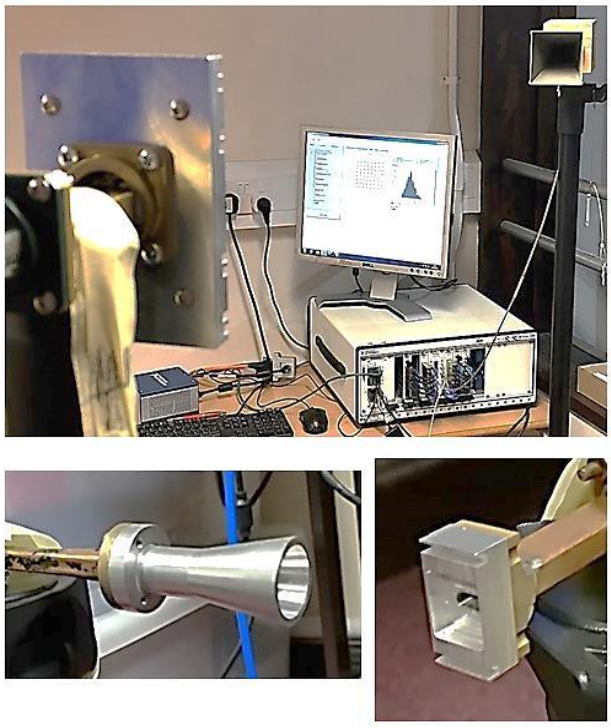


FIGURE 2.10: The mmWave hardware setup

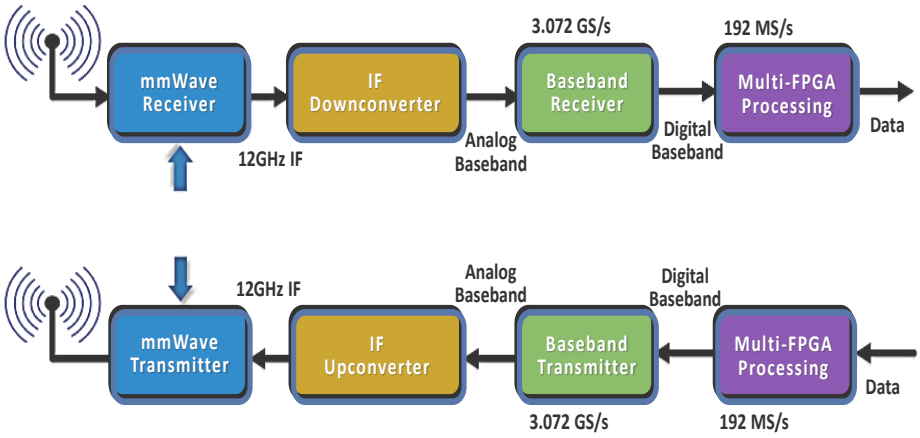


FIGURE 2.11: The mmWave system diagram [12]

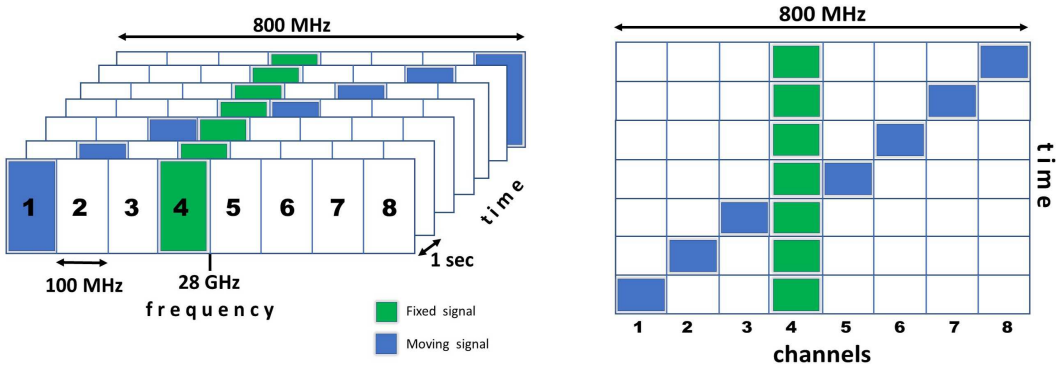


FIGURE 2.12: Representation of a dynamic pattern in the 800 MHz mmWave band with a fixed OFDM signal (the green rectangle) at channel 4 and a moving OFDM signal (the blue rectangle). 3D frequency-time diagram (left) and the corresponding 2D time-channels version (right)

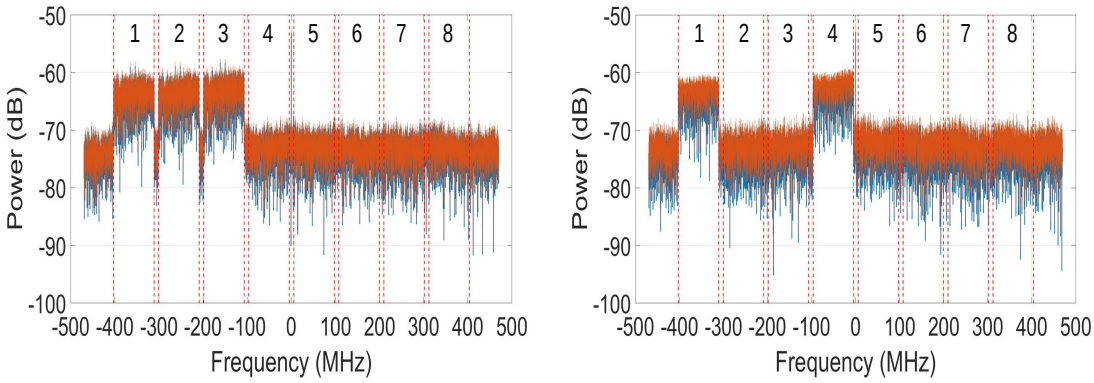


FIGURE 2.13: The mmWave spectrum at base-band: 8 x 100 MHz bandwidth. Occupied channels: 1^{st}, 2^{nd}, 3^{rd} (left) and 1^{st} and 4^{th} (right)

Fig. 2.10) for receiving the signal [4], respectively. The mmWave transceiver operates at 28 GHz (central carrier frequency) and the analysed spectrum consists of 8 *x* 100 MHz channels with 800 MHz total bandwidth, as shown in Fig. 2.12. The 8 channels are respectively found at the corresponding offset frequencies: -350, -250, -150, -50, +50, +150, +250, and +350 MHz with respect to the central frequency of the mmWave band and complex I/Q data is collected at base-band after the down-conversion process (see Fig. 2.13). Cyclic-Prefix Orthogonal Frequency Division Multiplexing (CP-OFDM) signals with 1200 sub-carriers are transmitted inside the mmWave band with 75 kHz sub-carrier spacing and 2048 FFT size. Different modulation schemes are supported (BPSK, QPSK, 16-QAM, and 64-QAM). Sampling frequency is 3.072 GS/s (12-14 bits).

Bibliography

[1] Implementing tv white spaces. office of communications (ofcom), february 2015 [online]. available: http://stakeholders.ofcom.org.uk/binaries/consultations/white-space-coexistence/statement/tvws-statement.pdf.

[2] N. Abbas, Y. Nasser, and K. El Ahmad. Recent advances on artificial intelligence and learning techniques in cognitive radio networks. *Eurasip Journal on Wireless Communications and Networking*, 2015, 12 2015.

[3] M. R. Akdeniz, Y. Liu, M. K. Samimi, S. Sun, S. Rangan, T. S. Rappaport, and E. Erkip. Millimeter wave channel modeling and cellular capacity evaluation. *IEEE Journal on Selected Areas in Communications*, 32(6):1164–1179, June 2014.

[4] S. Alkaraki, Y. Gao, and C. Parini. High aperture efficient slot antenna surrounded by the cavity and narrow corrugations at ka-band and ku-band. *IET Microwaves, Antennas Propagation*, 12(12):1926–1931, 2018.

[5] K. Dabčević, L. Marcenaro, and C. S. Regazzoni. Spd-driven smart transmission layer based on a software defined radio test bed architecture. In *4th International Conference on Pervasive and Embedded Computing and Communication Systems (PECCS) - Lisbon, Portugal*, January 2014.

[6] K. Dabčević, M.O. Mughal, L. Marcenaro, and C. S. Regazzoni. Spectrum intelligence for interference mitigation for cognitive radio terminals. In *Wireless Innovation Forum European Conference on Communications Technologies and Software Defined Radio (WInnComm- Europe), Rome, Italy*, November 2014.

[7] M. Elkashlan, T. Q. Duong, and H. H. Chen. Millimeter-wave communications for 5g: Fundamentals: Part i (guest editorial). *Communications Magazine, IEEE*, 52:52–54, 09 2014.

[8] A. Fiaschetti, J. Noll P., Azzoni, and R. Uribeetxeberria. *Measurable and Composable Security, Privacy, and Dependability for Cyberphysical Systems: The SHIELD Methodology*. CRC Press, 2017.

[9] Y. Gao, Z. Qin, Z. Feng, Q. Zhang, O. Holland, and M. Dohler. Scalable and reliable iot enabled by dynamic spectrum management for m2m in lte-a. *IEEE Internet of Things Journal*, 3(6):1135–1145, Dec 2016.

[10] O. Holland, S. Ping, A. Aijaz, J. Chareau, P. Chawdhry, Y. Gao, Z. Qin, and H. Kokkinen. To white space or not to white space: That is the trial within the ofcom tv white spaces pilot. In *2015 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, pages 11–22, Sept 2015.

[11] L. Hu, R. Shi, M. Mao, Z. Chen, H. Zhou, and W. Li. Optimal energy-efficient transmission for hybrid spectrum sharing in cooperative cognitive radio networks. *China Communications*, 16(6):150–161, June 2019.

[12] National Instruments. Introduction to the ni mmwave transceiver system hardware: http://www.ni.com/product-documentation/53095/en/, 2019.

[13] A. Kumar and S. Saha. Estimator-Correlator based Spectrum Sensing with PU Signal Uncertainty in Full Duplex CRNs. In *2019 URSI Asia-Pacific Radio Science Conference (AP-RASC)*, pages 1–4, March 2019.

[14] J. Li, Z. Feng, Z. Feng, and P. Zhang. A survey of security issues in cognitive radio networks. *China Communications*, 12(3):132–150, Mar 2015.

[15] S. Lin, L. Kong, Q. Gao, M. K. Khan, Z. Zhong, X. Jin, and P. Zeng. Advanced dynamic channel access strategy in spectrum sharing 5g systems. *IEEE Wireless Communications*, 24(5):74–80, October 2017.

[16] C. Liu, X. Liu, and Y. Liang. Deep CNN for Spectrum Sensing in Cognitive Radio. In *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, pages 1–6, May 2019.

[17] X. Liu and X. Zhang. NOMA-based Resource Allocation for Cluster-based Cognitive Industrial Internet of Things. *IEEE Transactions on Industrial Informatics*, pages 1–1, 2019.

[18] R. Lombardi. Millimeter-wave technology trends for 5g and wireless transmission applications and technologies. In *2017 IEEE MTT-S International Microwave Workshop Series on Advanced Materials and Processes for RF and THz Applications (IMWS-AMP)*, pages 1–1, Sep. 2017.

[19] Y. Ma, Y. Gao, Y. Liang, and S. Cui. Reliable and efficient sub-nyquist wideband spectrum sensing in cooperative cognitive radio networks. *IEEE Journal on Selected Areas in Communications*, 34(10):2750–2762, Oct 2016.

[20] J. H. Martin, L. S. Dooley, and K. C. P. Wong. New dynamic spectrum access algorithm for tv white space cognitive radio networks. *IET Communications*, 10(18):2591–2597, 2016.

[21] Z. Qin, Y. Gao, and C. G. Parini. Data-assisted low complexity compressive spectrum sensing on real-time signals under sub-nyquist rate. *IEEE Transactions on Wireless Communications*, 15(2):1174–1185, Feb 2016.

[22] Z. Qin, Y. Gao, M. Plumbley, and C.G. Parini. Wideband spectrum sensing on real-time signals at sub-nyquist sampling rates in single and cooperative multiple nodes. *IEEE Transactions on Signal Processing*, 64(12):3106–3117, June 2016.

[23] Z. Qin, Y. Gao, and M. D. Plumbley. Malicious user detection based on low-rank matrix completion in wideband spectrum sensing. *IEEE Transactions on Signal Processing*, 66(1):5–17, Jan 2018.

[24] Z. Qin, Y. Gao, M. D. Plumbley, and C. G. Parini. Wideband spectrum sensing on real-time signals at sub-nyquist sampling rates in single and cooperative multiple nodes. *IEEE Transactions on Signal Processing*, 64(12):3106–3117, June 2016.

[25] Y. Song, W. Yang, X. Yang, Z. Xiang, and B. Wang. Physical Layer Security in Cognitive Millimeter Wave Networks. *IEEE Access*, 7:109162–109180, 2019.

[26] S. Srinu, M. K. K. Reddy, and C. Temaneh-Nyah. Physical layer security against cooperative anomaly attack using bivariate data in distributed CRNs. In *2019 11th International Conference on Communication Systems Networks (COMSNETS)*, pages 410–413, Jan 2019.

[27] S. Wang, K. Huang, X. Xu, and S. Zhang. On the reliability and security performance of opportunistic relay selection in millimeter wave networks. In *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, pages 1–6, Aug 2018.

[28] W. Wang and Z. Zheng. Hybrid MIMO and Phased-Array Directional Modulation for Physical Layer Security in mmWave Wireless Communications. *IEEE Journal on Selected Areas in Communications*, 36(7):1383–1396, July 2018.

[29] Z. Wei, B. Zhao, and J. Su. Cooperative Sensing in Cognitive Radio Ad Hoc Networks. In *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, pages 1–6, May 2019.

[30] H. Xiao, D. Zhu, and A. T. Chronopoulos. Power Allocation With Energy Efficiency Optimization in Cellular D2D-Based V2X Communication Network. *IEEE Transactions on Intelligent Transportation Systems*, pages 1–11, 2019.

[31] D. Yang, G. Xue, J. Zhang, A. Richa, and X. Fang. Coping with a smart jammer in wireless networks: A stackelberg game approach. *IEEE Transactions on Wireless Communications*, 12(8):4038–4047, August 2013.

[32] W. Yang, L. Tao, X. Sun, R. Ma, Y. Cai, and T. Zhang. Secure on-off transmission in mmwave systems with randomly distributed eavesdroppers. *IEEE Access*, 7:32681–32692, 2019.

[33] X. Zhou, M. Sun, G. Y. Li, and B. Fred Juang. Intelligent wireless communications enabled by cognitive radio and machine learning. *China Communications*, 15(12):16–48, Dec 2018.

[34] Y. Zou, J. Zhu, X. Wang, and L. Hanzo. A survey on wireless security: Technical challenges, recent advances, and future trends. *Proceedings of the IEEE*, 104(9):1727–1765, Sept 2016.

3 Spectrum representation and feature extraction

In this chapter, some current work about signal detection and representation for spectrum sensing along with feature extraction from the detected signals is firstly introduced. Afterwards, non-stationarity and time-frequency analysis of dynamic signals is discussed. Two practical applications and current advances in these fields are presented in detail. This is the first and introductory stage **(stage 1)** close to radio environment and the R2D interface in the general cognitive capability diagram of Fig. 3.1.

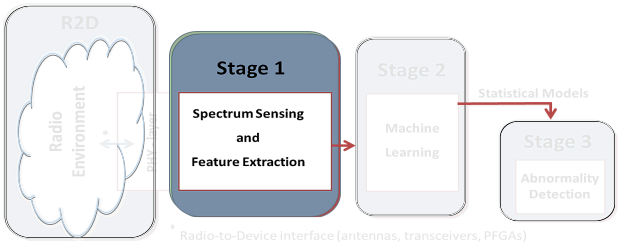


FIGURE 3.1: Spectrum sensing and feature extraction phase (stage 1)

3.1 Spectrum sensing and corresponding methods in the literature

In its initial phase, the CR was conceived to allow the unlicensed users or secondary users (SUs) to access the spectrum provided that SUs generate no or limited interference to licensed users or primary users (PUs) [2], [35]. This is the basic idea of the opportunistic spectrum access where the SUs are required to constantly perform *spectrum sensing (SS)* with the objective of obtaining awareness of the shared spectrum (also called situation-awareness) by gathering information about the surrounding channels. Many SS algorithms have been proposed in the literature. The estimator correlator detector (ECD) in [30] can be implemented to achieve optimal detection performance when the full knowledge of signal correlation and noise power is available. However, in most practical scenarios, information corresponding to the signal distribution is difficult to obtain and therefore, semi-blind spectrum sensing algorithms such as the energy detector (ED) [50], [18], [11], the maximum eigenvalue detector (MED) [58] and the generalized likelihood ratio test (GLRT) based signal

subspace eigenvalues (SSE) [59] algorithm have been proposed. The semi-blind algorithms only require the noise power information and are more widely applicable. In scenarios where the noise power is inaccurate or not available, the totally-blind spectrum sensing algorithms such as the information theoretic criteria (ITC) based methods [57, 52], the GLRT based arithmetic to geometric mean (AGM) [59], and maximum to average eigenvalue ratio (MAER) [51] algorithms are proposed.

However, in real scenarios in CR, non-stationary signals mainly compose the spectrum under analysis. If this is the case, time-frequency based approaches overcome the assumption of signal stationarity (like in Fourier transform) and provide analysis in both domains (time and frequency) to perform accurate SS.

In the following, after introducing some conventional techniques for SS, two recent works based on time-frequency representation of a wideband signal and a cyclostationary feature detector for interference mitigation, respectively, are discussed. The former can be used for SS in a dynamic spectrum, while the latter differentiates signals with different modulation schemes.

3.2 Conventional features and detectors

In the CR framework, there are many SS schemes such as Energy Detector, Matched Filter Detector, Cyclostationary Feature Detector, Wavelet-based SS.

- *Energy detector (ED)*: this process formally corresponds to solving the decision problem between the following two hypotheses [15]:

$$Z(n) = \begin{cases} \eta(n), & H_0 \\ S(n) + \eta(n), & H_1 \end{cases} ; \quad n = 1, ..., N_S \tag{3.1}$$

where $Z(n)$, $S(n)$ and $\eta(n)$ are the received signal, the transmitted signal and the noise samples, respectively. H_0 is the *null hypothesis* corresponding to the absence of the signal (in this case, received signal consists of only noise), and H_1 is the *alternative hypothesis* corresponding to the presence of the signal, while N_S is the number of samples acquired during the sampling process. Finding the appropriate threshold is the principal challenge of any energy detection scheme. The most common approaches are the Constant Detection Rate (CDR) and Constant False Alarm Rate (CFAR) detectors. Energy detection is easy to implement and does not require prior knowledge of signal parameters, but its performance is highly dependent on noise levels and interference.

- *Matched Filter Detector (MFD)*: the received signal is convolved with a conjugated time-reverse version of the transmitted signal and then compared with a particular threshold level to recover the transmitted signal without any error [54]. MFD is the optimal linear filter for maximizing the signal to noise ratio (SNR) in the presence of additive stochastic noise. Although in general this requires a non-flat frequency response, the associated distortion is not significant in situations such as radar and digital communications, where the original waveform is known and the objective is to detect the presence of this signal against the background noise. The main advantage of the matched filter detector is that it achieves both high processing gain due to coherency and good robustness to noise uncertainty with moderate computational complexity. On the other hand, it requires a priori knowledge of the primary user signal such as the modulation type and order, the pulse shape, and the packet format. If this information is not accurate, the matched filter performs poorly [31]. In addition, matched filtering detector requires a dedicated receiver structure which may not be possible in a practical CR terminal.

- *Cyclostationary Feature Detector (CFD)*: it exploits the cyclostationarity of modulated signals by detecting spectral peaks in spectral correlation function (SCF) [40]. The major advantage of CFD based detector lies on its abilities to perform better than energy detector at low signal-to-noise ratio (SNR) values and to distinguish different modulated signals. Furthermore, the cyclic spectral analysis has been used as a robust tool for signal classification when the carrier frequency and bandwidth information is unavailable. This performance is achieved at the cost of increased implementation complexity. A process $x(t)$ is said to be wide-sense cyclostationary with period T_0 if its mean $E[x(t)] = \mu_x(t)$ and autocorrelation $E[x(t)x^*(t+\tau)] = R_x(t,\tau)$ are both periodic with period T_0, in such case, they can be defined respectively as $\mu_x(t+T_0) = \mu_x(t)$ and $R_x(t+T_0,\tau) = R_x(t,\tau)$. The major benefit of spectral correlation is its insensitivity to background noise. Furthermore, different types of modulated signals (BPSK, AM, FSK, MSK, QAM, PAM) with overlapping power spectral densities have highly distinct SCFs.

- *Wavelet based detector*: wavelet transform (WT) is employed to characterize singularities and edges exhibited in the local singular structure of the PSD of a wideband signal $r(t)$, denoted by $S_r(f)$ in frequency. Edges in the spectrum correspond to the locations of frequency discontinuities $\{f_i\}_{i=1}^{N-1}$ to be identified. WT provides good precision to detect the occupied spectrum and to identify and locate spectrum holes in the signal spectrum even for faded signals [49]. Formally, $\Psi(f)$ is a wavelet smoothing function with a compact support, m vanishing moments and m times continuously

differentiable. Widely-used examples for $\Psi(f)$ include the Gaussian function and the perfect reconstruction filter bank (PRFB). The dilation of $\Psi(f)$ by a scale factor s is given by:

$$\Psi_s(f) = \frac{1}{s}\Psi\left(\frac{f}{s}\right) \tag{3.2}$$

where s takes values from powers of 2, i.e., $s = 2^j$, $j = 1, 2, \cdots, J$. Letting $*$ denote convolution, the continuous wavelet transform (CWT) of $S_r(f)$ is given by $\mathcal{W}_s S_r(f) = S_r * \Psi_s(f)$. It has been shown that the local extrema of the first derivative and the zero-crossings of the second derivative characterize the signal irregularities. The first-order and second-order derivatives of $S_r(f)$ smoothed by the scaled wavelet $\Psi_s(f)$ are derived in [49].

However, with the recent advent of artificial neural networks (ANNs), deep learning has been witnessed as an effective method to extract the inner pattern from a massive amount of data. Indeed, deep learning-based approaches have been proposed that shows increased detection performance with respect to conventional techniques.

3.3 Example of feature extraction: spectrum correlation in modulated signals *

Spectrum sensing information plays a key role in anti-jamming systems. This information may be used to detect potential jamming entities [37] and to take proactive measures, as the channel hopping strategy in [10], to ensure communication continuity and security [14]. Moreover, a history of observations can be maintained and used to devise more effective anti-jamming tactics.

In the technical literature, various spectrum sensing techniques have been proposed for CRs [26], such as energy detector (ED) [11], cyclostationary feature detector (CFD) [33], matched filtering detector (MFD) [54], and wavelet transform (WT)-based detector [60]. Among these methods, the CFD is capable of detecting the primary signal from the interference and noise even in very low signal-to-noise ratio (SNR) regions. This detection performance is achieved at the cost of an increased implementation complexity. Generally, an energy detector fails at low SNRs while a matched filtering detector requires a dedicated receiver structure which may not be possible in a practical cognitive radio terminal. CFD exploits the cyclostationarity of modulated signals by detecting spectral peaks in spectral correlation function (SCF) or spectral coherence function (SOF) [22, 44, 40], which are sparse in both angular

* Work published (see [3] in *List of publications and under review work*)

(f) and cyclic (α) frequency domain. Major advantage of CFD based detector lies on its abilities to perform better than energy detector at low SNR values and to distinguish different modulated signals. Furthermore, the cyclic spectral analysis has been used as a robust tool for signal classification when the carrier frequency and bandwidth information is unavailable [53, 34]. A comparison among the most common sensing methods (energy detection, cyclostationary, radio identification, match filtering, and waveform-based sensing) in terms of complexity and accuracy is made in [56]. Features-based algorithms require classification methods to evaluate the most probable class where the observed features belong to. Choosing the most suitable classification algorithm is not the only challenge to obtain satisfying performance and classification accuracy. Indeed, selection of features and the feature extraction algorithm play a fundamental role as they strongly influence the accuracy on the classified signals. Spectrum intelligence algorithms in [15] and [16] utilize hand-crafted features such as bandwidth, magnitude, and variance for each of the signals. The former employs an ED-based classifier with a hypothesis test, while the latter proposes a Bayes-based classifier. The main drawback related to these two approaches is that both the variability of the extracted features and noise prevent to obtain accurate classification especially at low SNR level.

At the sensing phase, data acquisition systems consisting of physical devices such as antennas, sensors and processing units are used as interface. Hardware architectures have been used in the literature to implement spectrum sensing in cognitive radio systems, such as the universal software radio peripheral (USRP) in a GNU-Radio framework in [39, 38]. While, to validate the proposed CSI algorithm, an SDR hardware platform, described in Sec. 2.3.1, has been employed to generate modulated signals in a specified band, namely 0-120 MHz which includes the Very High Frequency (VHF) band at 30-88 MHz. Alternatively, the Ultra High Frequency band (UHF) band can be selected. The testbed is remotely controlled and can be employed in on-line applications.

3.3.1 Data acquisition and processing

Acquisition of the wideband RF spectrum is performed periodically for the frequency band of interest. This may be done by taking either a quiet or an active approach, depending on the implementation of the architecture. A quiet approach implies that the radio is able to performing sensing simultaneously with transmitting/receiving, whereas in active sensing, the radio needs to stop transmitting/receiving while sensing takes place.

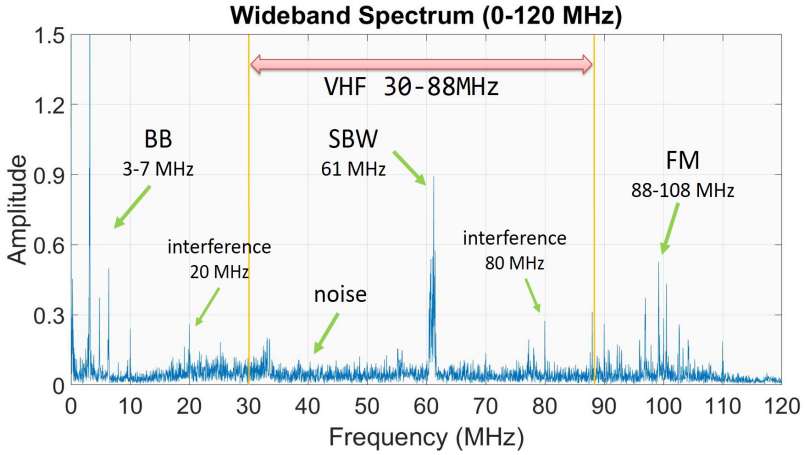


FIGURE 3.2: Spectrum observation in the 0-120 MHz band (1 burst) which includes the SBW signal at 61 MHz and with transmission power equal to -3dBm as well as other signals from external environment

Data Acquisition: exploiting the Selex testbed described in Sec. 2.3.1, real-data is collected and stored in a dataset. To this end, the VHF transmission band where the employed radios are operable has been utilized, meaning that the spectrum sensing is performed for the frequency band of 0-120 MHz.

The data consists of a large amount of spectrum observations containing a number of narrowband signals. More specifically, as shown in Fig. 3.2 (thin solid blue line), each spectrum consists of a SBW signal (digitally modulated signal) transmitted by the transmitting HH device and a number of signals (from the environment) such as the FM signal (in the 88-108 MHz band) and an unknown signal at 0-7 MHz. Two interference signals at 20 MHz and 80 MHz, respectively, are also observed. Further details are given in Sec. 3.3.2. The parameters of the SBW signal can be set remotely and, in particular, the transmission power and the carrier frequency of the transmitted SBW signal have been considered which are given 3 different values (7dBm-full, 4dBm-half, and -3dBm-one-tenth for the transmission power; 51MHz - 61MHz - 71MHz for the carrier frequency). Consequently, the dataset consists of spectrum measurements grouped in 9 different configurations, as shown in Tab. 3.1. Each configuration consists of more than 2500 bursts. The corresponding time-domain samples have also been stored in the dataset. The experimental step is discussed in Sec. 4.2.3.

Processing: this paragraph describes the pre-processing block of real-data to detect frequency bins belonging to the different waveforms inside the spectrum. The main parameters for the pre-processing are also described.

7dBm - *Full Power*		
I f_c = 51 MHz	II f_c = 61 MHz	III f_c = 71 MHz
4dBm - *Half Power*		
IV f_c = 51 MHz	V f_c = 61 MHz	VI f_c = 71 MHz
-3dBm - *Onetenth* Power		
VII f_c = 51 MHz	VIII f_c = 61 MHz	IX f_c = 71 MHz

TABLE 3.1: Configurations for the values of the transmission parameters (carrier frequency and transmission power) of the SBW signal in the collected dataset

First of all, the received spectrum observations are smoothed in the frequency domain through a simple moving average applied to the samples in order to reduce the sharp fluctuations due to noise which can be seen in each received spectrum.

Then, based on a sensible choice for a specific threshold, the background noise is eliminated, keeping only the FFT bins corresponding to actual signals. Basically, this process can be thought as an energy detection and formally corresponds to solving a decision problem between two hypotheses with a *null hypothesis* corresponding to the absence of the useful signal (in this case, received signal consists only of noise), and an *alternative hypothesis* corresponding to the presence of the useful signal. In most applications, the analysed spectrum is underutilised (usage of licensed bands is an example [6, 43]) which means that there is only a limited number of actual narrowband signals in the scanned wideband signal at any instant of time. In this scenario, suboptimal thresholding algorithms with low computational complexity can be considered where the threshold $\hat{\delta}$ is adaptively set based only on the mean value of the magnitudes of the scanned wideband signal [15], and given by:

$$\hat{\delta} = 2 \cdot \frac{1}{n} \sum_{n=1}^{N_S} |Z(n)| \tag{3.3}$$

Let K the number of frequency bins that are identified as a result of the thresholding process.

In a wideband and sparse spectrum observation there are L actual signals ($K > L$) and each of them involves a number of bins. For this reason, frequency bins corresponding to the same signal need to be grouped together and consecutive samples in the same group are classified as single waveform. After the thresholding step, grouped waveforms undergo smoothing, in order to reduce impacts of the imperfect

and erroneous sampling. For achieving this, a second stage moving average filter has been implemented.

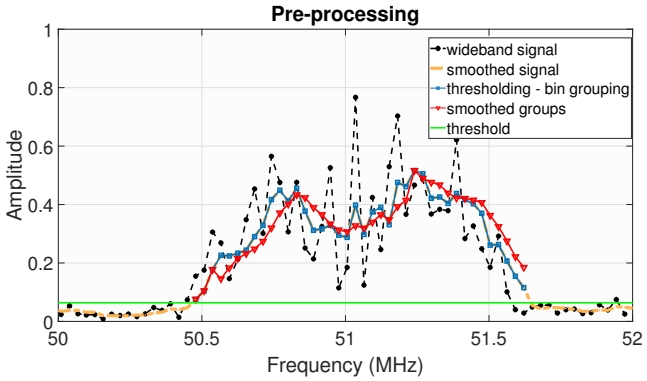


FIGURE 3.3: Pre-processing applied to the WB signal (in both the pictures, only the SBW signal is shown): (a) *WB signal*, *smoothed signal*, *thresholding and bin grouping*, and *threshold*; (b) also includes the waveform after *group smoothing*

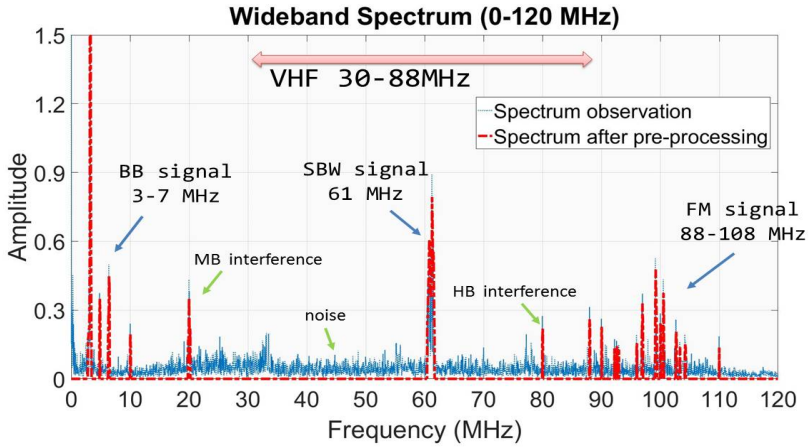


FIGURE 3.4: Wideband spectrum measurement before pre-processing (thin dotted blue line) and after pre-processing (thick dash-dotted red line)

Fig. 3.3 illustrates the difference between the original transmitted SBW signal, the corresponding smoothed signal, the sensed FFT bins, and the estimated signal after performing thresholding/bin grouping. The smoothed group corresponding to the SBW signal is also shown.

A comparison between the original wideband signal (thin dotted blue line) and the corresponding signal after having undergone the pre-processing (thick dash-dotted red line) is made in Fig. 3.4.

This concludes the frequency domain pre-processing phase which is applied to the collected wideband signals.

After this phase, the inverse Fourier transform is applied to produce the corresponding time domain signals from the pre-processed wideband spectrum measurements and, then, the CFD algorithm produces the α-profile from the time-domain sub-signals as described in Sec. 3.3.3.

3.3.2 Cyclostationary feature analysis

In this section, the classification process is described in order to present an application of the dataset to theoretical framework. After the pre-processing phase in the *Processing* paragraph above, the waveforms are classified as either potentially malicious (PM) or friendly (FR). The former class refers to signals which aim to disrupt or degrade communications among legitimate users (belonging to the latter class). In the 0-120MHz wideband spectrum, the jammer is supposed to be the SBW signal (which is capable of changing its transmission parameters), while legitimate waveforms as in Fig. 3.4 are the base-band (BB) signal, interference at 20 MHz (we call it medium-band, MB, interference) and 80 MHz (we call it high-band, HB, interference), and the FM signal. In order to differentiate them, a cyclostationary feature based algorithm with an ANN as classifier is applied to the dataset.

Cyclostationary feature analysis: when signal processing techniques assume communication signals as being stationary, their statistical parameters such as mean and variance do not vary with time. In this case, signals are modelled as one-dimensional (1D) autocorrelation function and the corresponding power spectrum density (PSD) can be obtained through Fourier transform of the autocorrelation function [4]. On the other hand, most of the signals in communication systems are in fact cyclostationary [23]. This means that their statistical parameters are periodically or cyclically stationary. A two-dimensional (2D) autocorrelation function is used to model cyclostationary signals where the additional dimension is the cycle frequency, denoted as α, at which the 1D autocorrelation function is computed. For each α, a cyclic-spectrum-cut is produced which is a function of the parameter α and denoted as $S_x^{\alpha=f_0}(f)$, namely a cyclic-spectrum-cut at $\alpha = f_0$. Considering the whole set of values for α, a three-dimensional (3D) cyclic spectrum function is obtained where the three dimensions are: f, α, and magnitude of the SCF. When $\alpha = 0$, the cyclic-spectrum-cut corresponds to the conventional PSD since the spectrum completely correlates with itself.

Specifically, a process $x(t)$ is said to be wide-sense cyclostationary with period T_0 if its mean $E[x(t)] = \mu_x(t)$ and autocorrelation $E[x(t)x^*(t+\tau)] = R_x(t,\tau)$ are

both periodic with period T_0, in such case, they can be defined respectively as:

$$\mu_x(t+T_0) = \mu_x(t)\,; \quad R_x(t+T_0,\tau) = R_x(t,\tau)\,. \tag{3.4}$$

The autocorrelation function of a wide-sense cyclostationary process can be expressed in terms of its Fourier series components:

$$R_x(t,\tau) = \sum_{\alpha} R_x^{\alpha}(\tau)\, e^{j2\pi\alpha t} \tag{3.5}$$

where $\alpha = \frac{a}{T_0}$ with a integer, $E[\cdot]$ is the expectation operator, α is the set of Fourier components, and $R_x^{\alpha}(\tau)$ represents the cyclic autocorrelation function (CAF) and gives Fourier components. CAF is given by:

$$R_x(\tau) = \lim_{T\to\infty} \frac{1}{T} \int_{\frac{-T}{2}}^{\frac{T}{2}} R_x(t,\tau) e^{-j2\pi\alpha t} dt\,. \tag{3.6}$$

When $R_x(t,\tau)$ is periodic in t with period T_0, (7) can be expressed as:

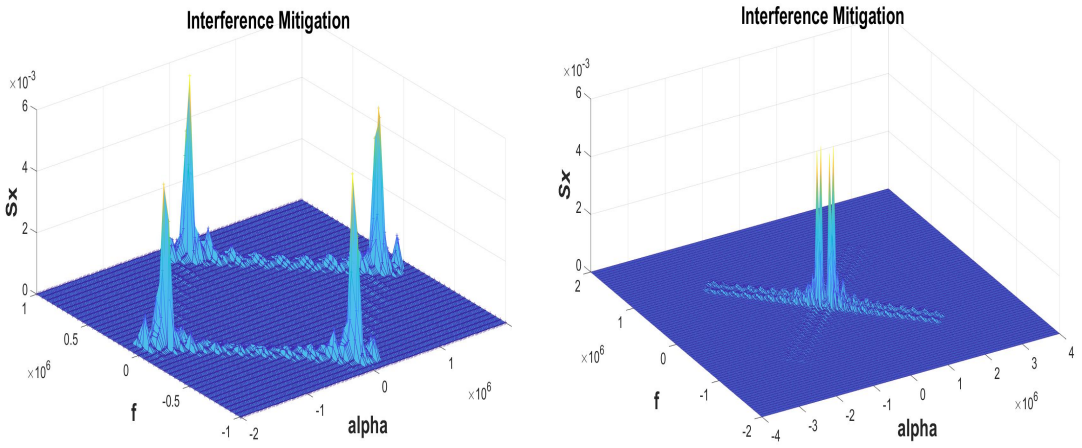
$$R_x^{\alpha}(\tau) = \frac{1}{T_0} \int_{\frac{-T_0}{2}}^{\frac{T_0}{2}} R_x(t,\tau) e^{-j2\pi\alpha t} dt\,. \tag{3.7}$$

The Fourier Transform of the CAF is known as SCF and is given by:

$$S_x^{\alpha}(f) = \int_{-\infty}^{\infty} R_x^{\alpha}(\tau) e^{-j2\pi f\tau} d\tau \tag{3.8}$$

where α is the cyclic frequency and f the angular frequency. The major benefit of spectral correlation is its insensitivity to background noise. Since the temporal correlation of different spectral components are measured, and the spectral components of noise are completely uncorrelated in time due to the fact that noise is wide-sense stationary process, such a noise does not play a significant role in the SCF. In other words, the existence of a correlation between separated spectral components, with separation equal to α, is called spectral redundancy [4] which is found in cyclostationary signals, while it is not usually observed in noise. This fact allows the spectral correlation of a signal to be accurately calculated even at low SNRs. Furthermore, different types of modulated signals - AM, FM, phase shift keying (PSK), frequency shift keying (FSK), minimum-shift keying (MSK), quadrature amplitude modulation (QAM), pulse-amplitude modulation (PAM), and so forth - with overlapping power spectral densities have highly distinct SCFs.

The SCF of the SBW signal and the 3-7 MHz BB signal in the 0-120 MHz spectrum are shown in Figs. 3.5(a) and 3.5(b), respectively.



(a) SCF of the 3-7 MHz BB signal (b) SCF of the SBW signal

FIGURE 3.5: SCF of two of the detected signals in the WB spectrum: (a) BB signal and (b) SBW signal

Since SCF computation requires large amount of data, which makes it unreasonable for a classifier to operate on it in real time, the cycle frequency profile (α-profile) has been employed in this work as a feature for classification. Specifically, the α-profile of SCF for a signal x is given by:

$$I(\alpha) = max_f[S_x^\alpha] \tag{3.9}$$

The α-profile of signals in the wideband spectrum of interest is shown in Sec. 4.2.3, while a more detailed description about SCF of some modulated signals can be find in Sec. 3.3.2 addressing the theoretical framework.

Cyclic spectral analysis: theoretical analysis of cyclic spectrum for a number of different modulation types (both analog and digital) can be found in the literature [20, 21, 24]. From the spectrum representation in Fig. 3.4, signals with different modulation schemes can be observed. For example, the BB signal found at low-frequency band 3-7 MHz is an AM signal, while the SBW is a digitally modulated signal with quaternary-PSK (QPSK) scheme. The two interferences are also digital modulated signals. Finally, the FM signal is observed on the right side of the spectrum. In the following, SCF of QPSK, AM, and FM signals is detailed as an example.

(QPSK) a PSK signal can be thought either as a binary amplitude-shift keying (ASK) for $M = 2$ or as a QAM when $M > 2$ (M is the number of points in the signal constellation), namely:

$$m(t) = u_I(t)\cos(2\pi f_0 t + p_0) - u_Q(t)\sin(2\pi f_0 t + p_0) \tag{3.10}$$

with in-phase and quadrature components $u_I(t)$ and $u_Q(t)$. f_0 is the carrier frequency and p_0 is a phase factor. In the specific case of QPSK signal, phases are separated by $\pi/2$ rad and the in-phase and quadrature components are time-aligned as follows:

$$u_I(t) = \sum_{n=-\infty}^{\infty} u_n^I q(t - nT_0 - t_0) \tag{3.11}$$

and

$$u_Q(t) = \sum_{n=-\infty}^{\infty} u_n^Q q(t - nT_0 - t_0) \tag{3.12}$$

respectively; the keying rate is defined by $1/T_0$, while t_0 is a fixed pulse-timing phase parameter. The carrier function $q(t)$ is a deterministic finite-energy pulse and modulated by u_n^I and u_n^Q which are random sequences assumed to be purely stationary. Consequently, the SCF can be written as [21]:

$$\begin{aligned} S_m^\alpha(f) = \frac{1}{2T_0} \Big[& Q(f + \alpha/2 + f_0) Q(f - \alpha/2 + f_0) S_{u_I}^\alpha(f + f_0) \\ & + Q(f + \alpha/2 - f_0) Q(f - \alpha/2 - f_0) S_{u_Q}^\alpha(f - f_0) \Big] e^{-i2\pi\alpha t_0} \end{aligned} \tag{3.13}$$

where the in-phase and quadrature components are assumed to be balanced such that $S_{u_I}^\alpha(f) - S_{u_Q}^\alpha(f) \equiv 0$. In this way, $S_{u_I}^\alpha(f) = S_{u_Q}^\alpha(f) = 1$ if $\alpha = k/T_0$, and $S_{u_I}^\alpha(f) = S_{u_Q}^\alpha(f) = 0$ if $\alpha \neq k/T_0$. The function $Q(f)$ is defined in [20] for a linear periodically time-variant system.

(AM) when a random lowpass signal $a(t)$ with PSD $S_a(f)$ is used to modulate the amplitude of a sine wave, the resulting amplitude modulated signal $m(t)$ is given by [25]:

$$m(t) = a(t) \cos(2\pi f_0 t + p_0) \tag{3.14}$$

and the general formula of its SCF is given by:

$$S_m^\alpha(f) = \begin{cases} \frac{1}{4} e^{\pm i2p_0} S_a(f), & \text{if } \alpha = \pm 2f_0 \\ \frac{1}{4} S_a(f + f_0) + \frac{1}{4} S_a(f - f_0), & \text{if } \alpha = 0 \\ 0, & \text{otherwise} \end{cases} \tag{3.15}$$

(FM) in many cases, the transmitted signal is modulated as:

$$m(t) = a \cos(2\pi f_0 t + \varphi(t)) \tag{3.16}$$

where $\varphi(t)$ is the phase of the FM signal, a is the carrier amplitude, and f_0 is the carrier frequency. The corresponding SCF is given by [55]:

$$S_m^{\alpha}(f) = \begin{cases} \frac{1}{4}\left[\Psi_r(f)\right] e^{\pm i2\varphi_0}, & \text{for } \alpha = \pm 2f_0 \\ \frac{1}{4}\left[\Psi_r(f+f_0) + \Psi_r(f-f_0)\right], & \text{if } \alpha = 0 \\ 0, & \text{otherwise} \end{cases} \tag{3.17}$$

where $\Psi_r(f)$ is the Fourier transform of the joint characteristic function for $\varphi(t+\tau/2)$ and $\varphi(t-\tau/2)$ given in [4].

Further details and SCF of other modulation schemes (both analog and digital) can be found in [4, 55, 20, 21, 25].

Complexity analysis (SCF computation complexity): efficient algorithms to compute the SCF are defined as FFT time smoothing algorithms. In this work, the strip spectral correlation algorithm (SSCA) has been considered. Basically, the SSCA is computed by multiplying the complex envelope with the conjugate of the received signal. The corresponding block diagram is discussed in [1] where 2 FFT blocks are employed. The complex envelope is a function of the frequency f, thus, the number of the first FFT points N' is inversely related to the frequency resolution Δf. While, the SCF formula is a 2D function in terms of f and α and the number of the second FFT points N is inversely proportional to the cyclic frequency resolution $\Delta\alpha$. This mean that an increase in the values of Δf and $\Delta\alpha$ will result in larger computational complexity although random effects are reduced and, consequently, the SCF reliability increases. The complexity of the SSCA algorithm is summarised in Tab. 3.2 in terms of the number of complex multiplications required to estimate the cyclic cross spectrum of two complex signals [1].

Computational section	Data tapering	N'-FFT
SSCA	$N'N$	$\frac{N'N}{2}\log_2 N$

Down-conversion	Sequences multiplication	N-FFT
$N'N$	$N'N$	$\frac{N'N}{2}\log_2 N$

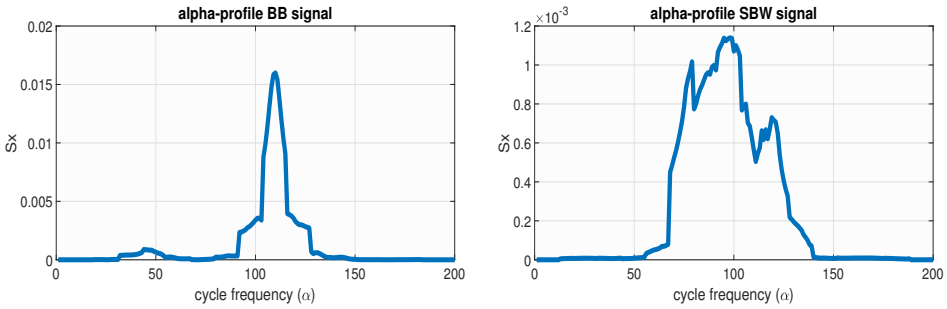
Table 3.2: Computational complexity for SSCA algorithm [1]

The total complexity is then given by $N'N\left(3+\frac{1}{2}\log_2 N'N\right)$. The SSCA is a highly parallel algorithm.

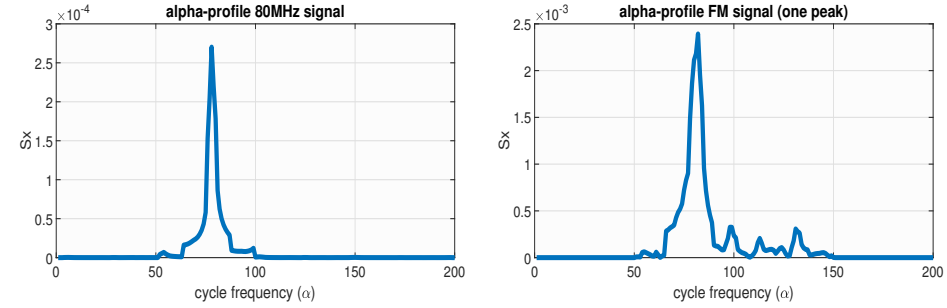
3.3.3 α-profile extracted from signals with different modulation schemes

Each of the detected waveform in the wideband spectrum, obtained after the pre-processing block in the *Processing* paragraph of Sec. 3.3.1, is characterized by its own α-profile which encompasses the cyclostationary feature. Specifically, each generated α-profile consists of 200 cyclic frequency points. Figs. 3.6 (a)-(d), show the α-profiles for 4 of the different signals detected in the 0-120 MHz band: the 3-7 MHz signal, SBW signal, interference at 80 MHz, and one peak of the FM signal.

It is worth noticing, that the frequency content of a single cyclostationary signal is mainly characterized by both its fundamental frequency and its cyclic frequency (the former is usually higher than the latter for RF signals). It can be expected that the cyclic frequency is independent of the fundamental frequency; consequently, the shape of the α-profile of that signal does not change with the fundamental frequency and the classifier is able to classify the signal independently of the part of the spectrum in which it is detected, as shown in Sec. 4.2.3. Future work will analyse signals located at RF, TVWS or ISM (industrial, scientific, and medical) band, for example, and the effective applicability of the proposed approach to signals in a spectrum beyond 120 MHz could be demonstrated.



(a) 3-7 MHz signal - BB (b) SBW signal



(c) 80 MHz signal - HB (d) one peak of the FM signal

FIGURE 3.6: α-profiles extracted from four different detected signals in the WB spectrum: a) baseband signal, b) SBW signal, c) interference at 80 MHz, d) one peak of the FM signal

This kind of feature is then used to differentiate friendly signals from potential malicious signals by learning a neural network; this is shown in Sec. 4.2 where the network acts as a binary classifier.

3.4 Non-stationarity and time-frequency analysis

The capability of describing the frequency content of a signal is a critical objective in diverse fields of science. Although the conventional Fourier Transform (FT) is an extremely important signal and image analysis tool, it assumes that a signal is stationary, i.e., that the frequency content is constant at all times in a signal, or at all locations in an image. However, in real applications like speech processing, geology, astronomy, or medicine, signals are non-stationary and frequency changes over time or space [9].

A theoretical background on non-stationarity is provided in [26] where the authors highlight that non-stationarity is an inherent characteristic of most, if not all, of the stochastic processes encountered in practice. A sample function of a complex continuous stochastic process is denoted by $x(t)$ and assumed to be harmonizable. Its mean function is $\mu_x(t) = \mathbf{E}[x(t)]$, while the covariance function is defined by $\Gamma_x(t_1, t_2) = \mathbf{E}[x(t_1)\, x^*(t_1)]$, the asterisk denotes complex conjugation. If the process is stationary, then, by definition, the covariance depends only on the time difference $\tau = t_1 - t_2$, namely $\Gamma_x(t_1, t_2) = \Gamma_x(t_1 - t_2)$. The *dynamic spectrum* is then defined as:

$$D(t_0, f) = \int_{-\infty}^{+\infty} \exp(-2j\pi\tau f)\, \mathbf{E}\left[x\left(t_0 + \frac{\tau}{2}\right) x^*\left(t_0 - \frac{\tau}{2}\right)\right] d\tau \tag{3.18}$$

where $t_0 = (t_1 + t_2)/2$. A more detailed description can be found in [26].

Consequently, a series of techniques have been developed to analyse dynamic signals and extract time-information about their spectral content by representing them as functions of both time and frequency. These methodologies are the foundation of time-frequency analysis and the corresponding signal representations are commonly referred to as time-frequency distributions (TFDs).

One of the methods for a non-stationary signal analysis is a decomposition of a signal into a set of blocks which can extract signal properties in time as well as in frequency [32]. This decomposition for a signal $x(t)$ can be written as:

$$x(t) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \lambda_x(\tau, \omega)\, h_{\tau,\omega}(\tau - t)\, d\tau d\omega \tag{3.19}$$

where the function $h_{\tau,\omega}(\tau)$ plays a role of a time-frequency atom which possesses joint time-frequency localization properties. The inverse transform of Eq. 3.19 is given by:

$$\lambda_x(t,\omega) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} x(\tau)\, h^*_{\tau,\omega}(\tau - t)\, d\tau \tag{3.20}$$

and $\lambda_x(t,\omega)$ can be interpreted as a linear time-frequency representation of $x(t)$. There is obviously a great arbitrariness in the choice of such a representation according to the choice of $h_{\tau,\omega}(\tau)$, providing several different TF transforms like the following ones. In general, many useful representations can be generated from the Cohen's class [13] by a specification of the kernel function.

Since TF operates in two-dimensional space, both time and frequency resolutions have to be considered and such resolutions depend on the window choice. Time and frequency resolutions cannot be improved simultaneously: when the frequency resolution increases, the time resolution decreases, as a consequence of the time-frequency uncertainty principle [36]. Several TF approaches are here described.

- *Short Time Fourier Transform (STFT)*: it is the most basic form of linear time-frequency transform [3] and defined as follows:

$$STFT(t,\omega) = \int x(\tau)\, h(\tau - t)\, e^{-j\omega\tau} d\tau \tag{3.21}$$

where $x(t)$ is the signal to be analysed, and $h(t)$ is the window function which is a symmetric function and is utilized to select a certain time interval of the signal. Changes in frequency over time are captured by sliding the window function to provide time localization. Due to the aforementioned time-frequency uncertainty principle, high frequency resolution is obtained with the window function as long as possible, which contradicts the time domain resolution. The commonly used window functions include rectangular window, triangular window, Hanning window, Hamming window, and Gauss window (in Gabor transform).

The discrete version of the STFT distribution can be found in [28]:

$$X(n,k) = \sum_{m=-\infty}^{+\infty} x[m]\, h[n-m] \exp^{-j2\pi km/N} R_N[k] \tag{3.22}$$

where $R_N[k] = u(k) - u(k - N + 1)$.

- *Wigner Ville Transform (WVT)*: another fundamental distribution in the TF analysis is the Wigner-Ville distribution [8]. It is defined as the Fourier transform of a local

correlation function called bilinear product [8] (indeed, unlike linear time-frequency transform, the signal appears twice in bilinear time-frequency transform):

$$WVT\left(t,\omega\right)=\int_{-\infty}^{+\infty}x\left(t+\frac{\tau}{2}\right)x^{*}\left(t-\frac{\tau}{2}\right)e^{-j\omega\tau}d\tau \tag{3.23}$$

where $x^{*}\left(t\right)$ is the conjugate of the signal $x\left(t\right)$.

Since time resolution and frequency resolution of WVD are independent to each other, WVD shows high aggregation properties. On the other hand, even though this distribution provides improved resolution over a single-window spectrogram, it suffers from cross-terms due to its bilinear nature which can clutter the corresponding TF signature [29]. Indeed, for multi-component signals, cross-terms are introduced, and the SNR is reduced. The more the multi-components are, the more the cross-terms are generated. To reduce the cross-terms, the Cohen distribution introduced a kernel function $g\left(\theta,\tau\right)$ to perform as a two-dimensional low-pass filter [13]. There are many kernel functions such as Born-Jordan [7] and Choi-Williams [12].

The discrete WVD of the time series $x\left(n\right)$ is given by:

$$DWVT\left(n,k\right)=\sum_{m=0}^{N-1}x\left(n+m\right)x^{*}\left(n-m\right)e^{-j2\pi mk/N} \tag{3.24}$$

where n and m are time indexes, while k denotes the frequency index, and N is the number of samples in the discrete time series.

- *Wavelet Transform (WT)*: it improves on the STFT by introducing the concept of progressive resolution [17]. Transforms that incorporate progressive resolution are known to provide better, more consistent time-frequency representations across the entire spectrum. The WT provides the equivalent of finer time resolution at high frequencies and finer frequency resolution at low frequencies. However, the WT does not measure frequency but only an analogue, called scale. Additionally, the WT provides either no phase information, or phase measurements which are all relative to different local references. This is in contrast to the conventional concept of phase, as provided by the FT, where all phase measurements are relative to a global reference.

Wavelet is a waveform function $\Psi\left(t\right)$ with limited support in time and zero average which indicates that it is an oscillating function. Wavelets are not periodic and may have discontinuous derivatives; signals with rapid changes are analysed better with the non-periodic wavelets. The WT is defined as the projection of the time signal $x\left(t\right)$ onto a set of functions $\{\Psi_{s,d}\left(t\right)\}$ (daughter wavelets) obtained by translating and

scaling the original wavelet $\Psi(t)$ (mother wavelet). They determined as:

$$\Psi_{s,d}(t) = \frac{1}{\sqrt{s}}\Psi\left(\frac{t-d}{s}\right) \tag{3.25}$$

where s is the positive adimensional scale parameter and d is the delay. The scale parameter stretches or compresses the mother wavelet and is connected to the frequency (low scale gives compressed wavelet and thus it can better analyse rapidly changing features what means high frequency components of a signal and vice versa). The delay parameter shifts the wavelet along the time axis and is connected to the time. Both parameters vary continuously. The WT of the signal is then:

$$W^x(s,d) = \int_{-\infty}^{+\infty} x(t)\,\Psi^*_{s,d}(t)\,dt = \int_{-\infty}^{+\infty} x(t)\,\frac{1}{\sqrt{s}}\Psi^*_{s,d}\left(\frac{t-d}{s}\right)dt \tag{3.26}$$

where wavelet coefficients $\{W^x(s,d)\}$ are functions of scale and delay. The basis of WT is not unique and should be chosen according to the characteristics of the signal. Time and frequency resolutions depend on the wavelet choice. There are a number of mother wavelets that can be used for analysis such as Gabor mother wavelet which has the best time-frequency resolution.
The discrete version of the wavelet transform can be found in [19].

- *Stockwell Transform (ST)*: developed by Stockwell et al. [47] and based on a sliding Gaussian window distribution. It exhibits globally referenced phase and frequency measurements similar to those of the STFT, as well as the progressive resolution of the WT. This combination of desirable features makes ST particularly suitable in several fields, such as biomedical signal and image analysis applications, and has shown promise in one of the most recent applications which has been employing the ST extensively, namely the Automatic Modulation Recognition (AMR) [45]. Despite this, the computational demands of the ST, due to redundant representation of time-frequency, in some cases limit its utility and prevent more widespread usage.

The ST of a non-stationary time signal $x(f)$ is defined as:

$$S(\tau,\nu) = \int_{-\infty}^{+\infty} x(t)\,\frac{|\nu|}{\sqrt{2\pi}}e^{-\frac{(\tau-t)^2\nu^2}{2}}e^{-i2\pi\nu t}dt \tag{3.27}$$

where τ and ν are the transform time and frequency coordinates. This equation has the same form as that of the FT, but adds a normalized-area Gaussian window for time localization. The center of the window is τ, and the variance of the window is $1/\nu^2$, consequently the width of the window decreases with increasing frequency. This automatically adjusts the ST window to provide progressive resolution.

The ST of a discrete time series $\{r[p]\}$, given by [5], is discussed in Sec. 3.5.2.

- *Hilbert-Huang Transform (HHT)*: a multi-component non-stationary signal is decomposed by using the empirical mode decomposition (EMD) technique [27] and, then, the Hilbert Transform (HT) is applied to the obtained components known as intrinsic mode functions (IMFs). Although the decomposition is adaptive and therefore highly efficient, IMFs suffer from mode mixing issue that results in an improper time-frequency representation.

From time-frequency analysis described in this section, features for each signal such as the central frequency, bandwidth, and transmitting power can be extracted and used as input data-set for the learning step to estimate model parameters for each signal, transition probability, and interaction as discussed in Sec. 4.5.

3.5 High sampling rate dynamic signals and ST *

ST is a time-frequency technique which provides time-information of the spectral content of the signal by observing it through a sliding Gaussian window. A major drawback of ST is heavy computational cost [42] especially in wideband applications where high sampling rate signals with non-stationary behaviour have to be represented.

This work introduces a dual-resolution (DR) approach based on discrete ST which reduces the computational time of the conventional ST by increasing time delay of the sliding window. Specifically, the vector signal is partitioned into sub-blocks in which the wideband signal is assumed to be locally stationary and the sliding window is moved on the first sample of each sub-block. This results in a control capability on both time-resolution and computational load that allows a dynamic trade-off to be adaptively selected.

Time-frequency analysis can provide information to analyse dynamically adapted transmitter spectrum occupancy strategies.

In this context, the DR technique is conceived to deal with dynamic wideband signals sampled at high sampling rate and to produce a better trade-off between time-resolution and computational time. The proposed approach has been validated in a dynamic scenario consisting of real data collected with a software defined radio testbed. As described in Sec. 3.5.5, the main target for this work is signal identification in Cognitive Radio (CR) through features extracted from each detected signal in the time-frequency representation such as bandwidth, carrier frequency, amplitude,

* Work published (see [1] in *List of publications and under review work*)

and variance. To this end, a comparison between the ST representation and the corresponding STFT of the analysed spectrum supports the choice of using ST.

The remaining part of this section is organised as follows: Sec. 3.5.1 introduces the time-frequency framework; discrete ST and the DR approach are described in Sec. 3.5.2; the testbed and data acquisition are in Sec. 3.5.3; while the validation and comparison with STFT are detailed in Sec. 3.5.4; conclusion and some future directions are in Sec. 3.5.5.

3.5.1 Problem formulation

In this work, a DR approach applied to ST is proposed and the corresponding representations are compared with the ones obtained through STFT. The latter is computed by applying the discrete Fourier transform (DFT) through a fixed-sized, moving window to a given time series $\left\{r_p\right\}_{p=0}^{P-1}$ of finite length P [41]. The window is moved by one time point at time resulting in overlapping windows. For a rectangular window of length N, the STFT representation consists of N frequency points and $P-N+1$ time points. The implementation of the STFT matrix can be found in [41]. Compared with the ST, STFT exhibits less computational complexity. However, as shown in Sec. 3.5.4, the corresponding representations are not sufficient for possible application to signal identification through features extracted from detected waveforms.

3.5.2 Discrete ST and the dual-resolution method

The main objective of this section is to introduce the proposed approach for the discrete model of the ST distribution. More specifically, $\{r\left[p\right], p=0,1,\cdots,P-1\}$ denotes the discrete time series corresponding to a continuous signal $r\left(t\right)$ with a time sampling interval T. The discrete ST of $\{r\left[p\right]\}$ is given by [5]:

$$S_T\left[m,n\right]=\sum_{p=0}^{P-1} r\left[p\right]\frac{|n|}{\sqrt{2\pi}kN}e^{-\left(n^2(m-p)^2/2k^2N^2+j2\pi pn/N\right)} \tag{3.28}$$

when $m=0,\cdots,M-1;\ n=1,\cdots,N-1$; and by:

$$S_T\left[m,0\right]=\frac{1}{P}\sum_{p=0}^{P-1} r\left[p\right] \tag{3.29}$$

when $n=0$; m is the time delay of the sliding window, n denotes the index of frequency range, p denotes the time index, and k is a scaling factor that controls the time-frequency resolution. When k increases, the frequency resolution increases, with a corresponding loss of time resolution [46].

When $m = 0, \cdots, M-1$ and $n = 0, \cdots, N-1$ (with $M = N = P$ from now on), the number of time and frequency samples generated by the ST is exactly $P \cdot P$ and, consequently, the computational load is limited by the number of the samples, P, in the signal. Considering that the parameter m is the time delay of the sliding window over the signal $r[p]$ and assuming that the signal is locally stationary within the time corresponding to several time delays m, a different approach is proposed in this work which reduces the computational load of the discrete ST. From Fig. 3.7, the P samples of the locally stationary signal are divided into s sub-blocks each consisting of P/s samples.

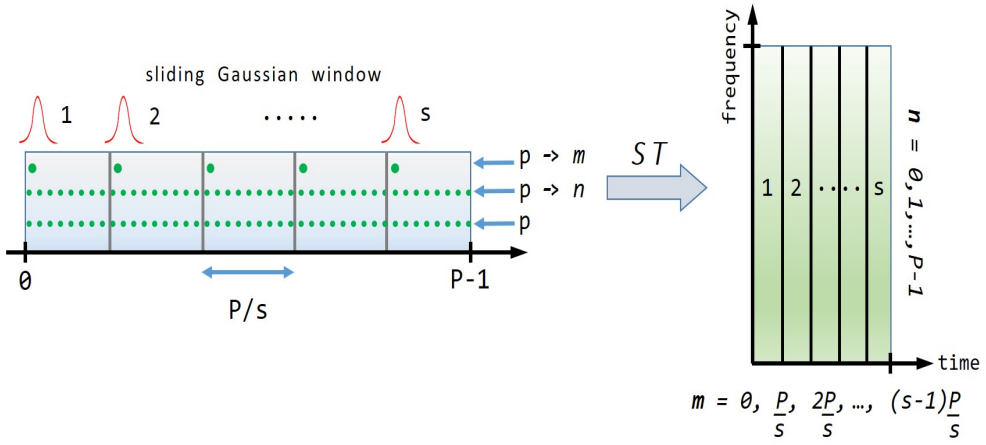


FIGURE 3.7: Dual-resolution technique based on ST

This approach computes the ST on s equally spaced values for m belonging to the set $\{0, \cdots, P-1\}$, while $n = 0, \cdots, P-1$ remains as before. Indeed, since in this way the time delay of the sliding window is increased, all the values of m in each sub-block (unless the first one) are not directly included in the computation of the ST (it is sufficient to increase the sliding window length, by increasing k, to also cover the discarded samples). This is based on the consideration that the signal is locally-stationary over P/s m's, or (P/s)-stationary, namely variations of the parameters happen no faster than the time corresponding to P/s samples and not necessarily at the boundary between two consecutive sub-windows. For quicker changes, the length of the sub-blocks can be reduced by increasing s accordingly. This introduces a new capability of the ST, namely a trade-off between time-resolution and computational time that can be adaptively controlled through s, P, and k. The frequency resolution can be changed by regulating P or k (or both).

Eqs. (3.28) and (3.29) can be written in a vector form as follows:

$$S_T[m,n] = \mathbf{T}_{mn}\mathbf{r} \tag{3.30}$$

where the $P \times 1$ vector $\mathbf{r} = [\, r[0], r[1], \cdots, r[P-1] \,]^T$ collects the samples of the discrete series, while the following $1 \times P$ vector

$$\mathbf{T}_{mn} = [\, T_{mn,0}, T_{mn,1}, \cdots, T_{mn,(P-1)} \,]$$

consists of elements given by [5]:

$$T_{mn,p} = \frac{|n|}{\sqrt{2\pi} k N} e^{-\left(n^2 (m-p)^2 / 2k^2 N^2 + j2\pi pn/N\right)} \tag{3.31}$$

$p = 0, \cdots, P-1$. By using the $(s \cdot N) \times P$ transform matrix

$$\mathbf{T} = \Big[\, \mathbf{T}_{00}^T, \cdots, \mathbf{T}_{0(N-1)}^T, \mathbf{T}_{\frac{P}{s}0}^T, \cdots \\ \cdots, \mathbf{T}_{\frac{P}{s}(N-1)}^T, \cdots, \mathbf{T}_{\frac{P(s-1)}{s}0}^T, \cdots, \mathbf{T}_{\frac{P(s-1)}{s}(N-1)}^T \,\Big]^T$$

the discrete ST can be modeled as a linear vector equation:

$$\mathbf{s} = \mathbf{T}\mathbf{r} \tag{3.32}$$

in which the $(s \cdot N) \times 1$ vector $\mathbf{s}$ consists of elements s_{mn} corresponding to $S_T[m, n]$. In this work, the dimensionality of Eq. (3.32) is drastically reduced because $m = m_s P / s$ where $m_s = 0, \cdots, s-1$ is the delay index, instead of $m = 0, 1, 2, \cdots, P-1$. The $\mathbf{T}$ matrix in Eq. (3.32) can be generated through the following Matlab code:

```
clear;
s = 64; k = 17.5;
N = 512;        % frequency samples in the ST
M = 512;        % time samples in the ST
step = M/s;
P = 512;        % samples in the time series
m_ = 0:step:M-1;  n_ = [0,1:N-1];  % time axis and frequency
    axis in the ST
T0 = abs(n_)/(sqrt(2*pi)*k*N);  T0(1) = abs(sqrt(2*pi)*k)/(sqrt
    (2*pi)*k*N);

T = zeros(s*N,P);  Tr = zeros(s*N,P);  Ti = zeros(s*N,P);
for p = 1:P
    vr_ = ((m_-p).^2)/(2*k^2*N^2); % real part
    vi_ = 1i*2*pi*p/N;  vi_ = conj(repmat(vi_,1,s)');  % imag
        part

    Br = zeros(s,N);  % real part of the exponent
    for t = 1:N
```

```
        Br(:,t) = vr_'*n_(t)^2;
    end
    Br_exp = exp(-Br);  % real part
    Tr(:,p) = reshape(Br_exp,[],1);

    Bi = zeros(s,N);  % imag part of the exponent
    for t = 1:N
        Bi(:,t) = vi_*n_(t);
    end
    Bi_exp = exp(-Bi);  % imag part
    Ti(:,p) = reshape(Bi_exp,[],1);

    T(:,p) = Tr(:,p).*Ti(:,p);    % T matrix
    for t = 1:N
        T((t-1)*s+1:t*s,p) =  T0(t)*T((t-1)*s+1:t*s,p);
    end
end
```

Now, let B the frequency bandwidth of the signal of interest, which in practical cases is from several MHz to GHz for wideband signals. Consequently, the sample rate is in the order of $2B$, or slightly more. In the following, (P/s)-stationarity of the signal means that is should be stationary in a time length of $\frac{P/s}{2B}$ secs. In Sec. 3.5.4, the values of the ratio P/s are from 4 to 64. Consequently, $\frac{P/s}{2B}$ is from about 16 to 267 $n$secs for $B = 120$ MHz. While, it is from 2 to 32 $n$secs, when $B = 1$ GHz. Although the piecewise assumption is a limitation of the DR approach, the time interval in which the signal should be stationary is very short and compatible with real systems.

Concerning the frequency resolution defined as $R = B/P$ (which should be a small value), P must be as large as possible for wideband signals in order to have reasonable resolution. This motivates the DR approach.

3.5.3 Data Acquisition of real dynamic signals

By means of the testbed described in Sec. 2.3.1, real-data is collected in the frequency band 0-120 MHz. As shown in Fig. 3.8 (thin dotted blue line), each received spectrum consists of a digitally modulated SBW signal in the VHF band (from the transmitting HH) and a number of signals (from the environment) such as the FM signal (in the 88-108 MHz band) and the baseband (BB) signal at 0-7 MHz, in addition to interference (at 20 and 80 MHz) and noise. The SBW signal is capable of hopping within the VHF band.

The intermediate step before ST analysis is pre-processing in the frequency domain which is applied to the data in order to: reduce the fluctuations in the spectrum

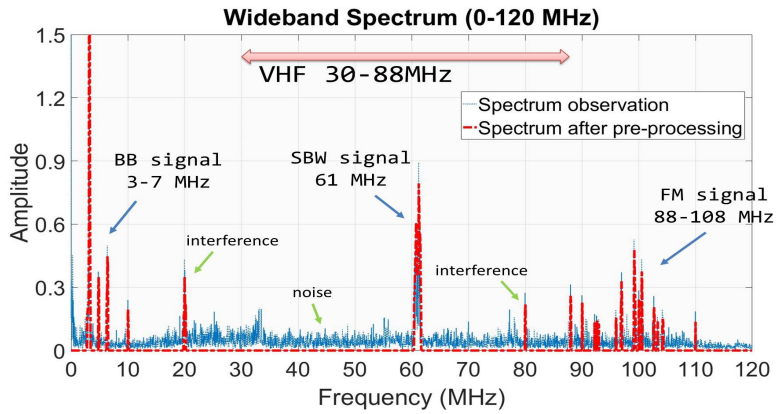


FIGURE 3.8: Wideband spectrum, which consists of a number of signals including the SBW at 61 MHz, before pre-processing (thin dotted blue line) and after pre-processing (thick dash-dotted red line)

by smoothing, remove the noise and detect the signals through energy detector, group the FFT bins which belong to the same signal, and smooth the grouped bins [16]. The resulting spectrum observation is shown in Fig. 3.8 (thick dash-dotted red line).

3.5.4 Validation and comparison with STFT

To validate the proposed dual-resolution approach, the discrete ST has been applied to real data to obtain time and frequency representation. Specifically, at four different time instants the carrier frequency of the transmitted SBW signal assumes sequentially the values 41 - 51 - 61 - 71 [MHz], while the transmit power is 7dBm. The other waveforms extracted from the spectrum in Fig. 3.8 include: 4 sub-signals close to one other (spacing about 0.5-1 MHz) at low frequencies, 2 interference signals at 20 MHz and 80 MHz, respectively, and 3 sub-signals in the FM band (spacing about 2.5 and 7 MHz). The carrier frequency of these waveforms is fixed over the measurement time.

According to Tab. 3.3, the analysed parameters are P (number of samples in the signal to be S-transformed), s (number of sub-blocks), and the scaling factor k defined in Sec. 3.5.2. The SBW signal is at least (P/s)-stationary for each configuration.

P				s			k		
256	512	640	1024	16	32	64	5	17.5	25

TABLE 3.3: Parameters used to validate the dual-resolution approach

Validation is performed through Matlab® 2106b. Fig. 3.9(a) is the conventional ST, obtained with $k = 17.5$ and without DR, of a signal which consists of $P = 512$

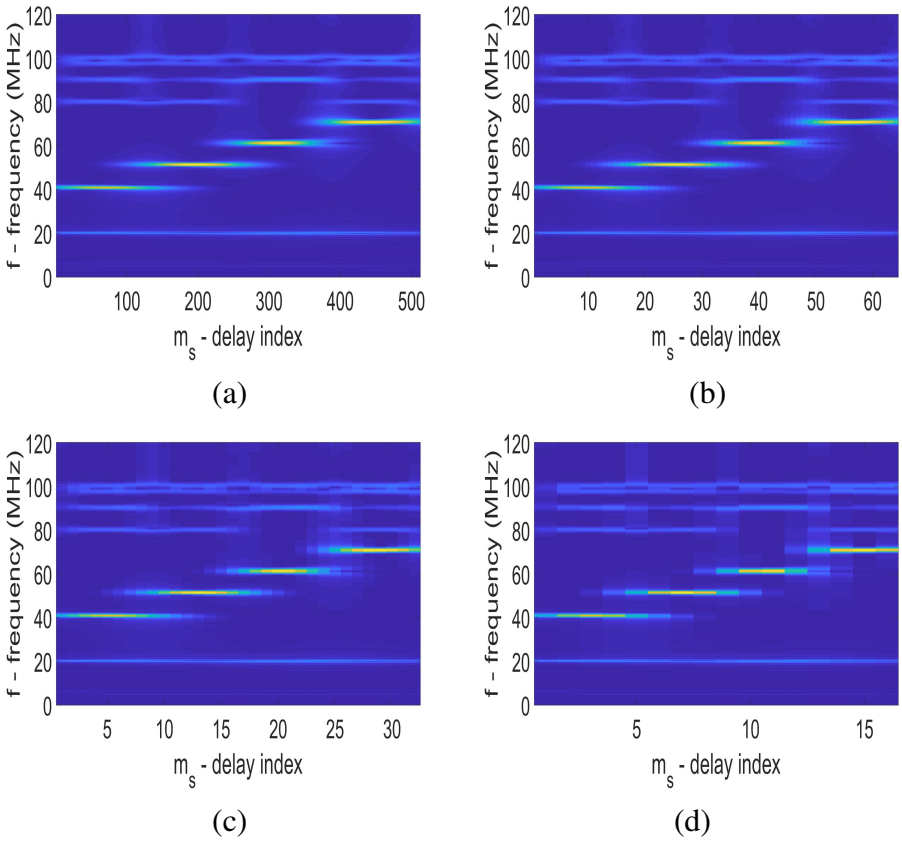


FIGURE 3.9: ST representation of the wideband signal with $P = 512$ and $k = 17.5$: a) without DR; and with dual-resolution where b) $s = 64$, c) $s = 32$, and d) $s = 16$. The SBW jumps among 41-51-61-71 MHz at different time instants

samples. The parameter m_s is the delay index as defined in Sec. 3.5.2, namely the time variable after ST. The waveforms in the spectrum can be clearly distinguished. In particular, the SBW signal in the middle of the figure jumps sequentially to the 4 different frequencies at 4 consecutive time instants. At the bottom, the BB peaks are visible (they are very close to each other; the progressive resolution of ST produces fine frequency resolution at low frequencies [9]). Just above them, there is the interference at 20 MHz. Beyond the SBW signal, the interference at 80 MHz and the 3 FM sub-signals can be seen (the progressive resolution of ST produces low frequency resolution at high frequencies [9]). In this case, the amount of frequency-time samples is $P \cdot P$. The dual-resolution technique generates the ST representation in Figs. 3.9(b), 3.9(c), and 3.9(d) in which s is 64, 32, and 16, respectively. Although the time resolution gets worse by reducing the number of sub-blocks, the signals in the spectrum remain clear. In particular, the dynamic hopping of the SWB signal can be still observed. The main advantage is that the amount of frequency-time samples is reduced to $P \cdot s$, with $s \ll P$.

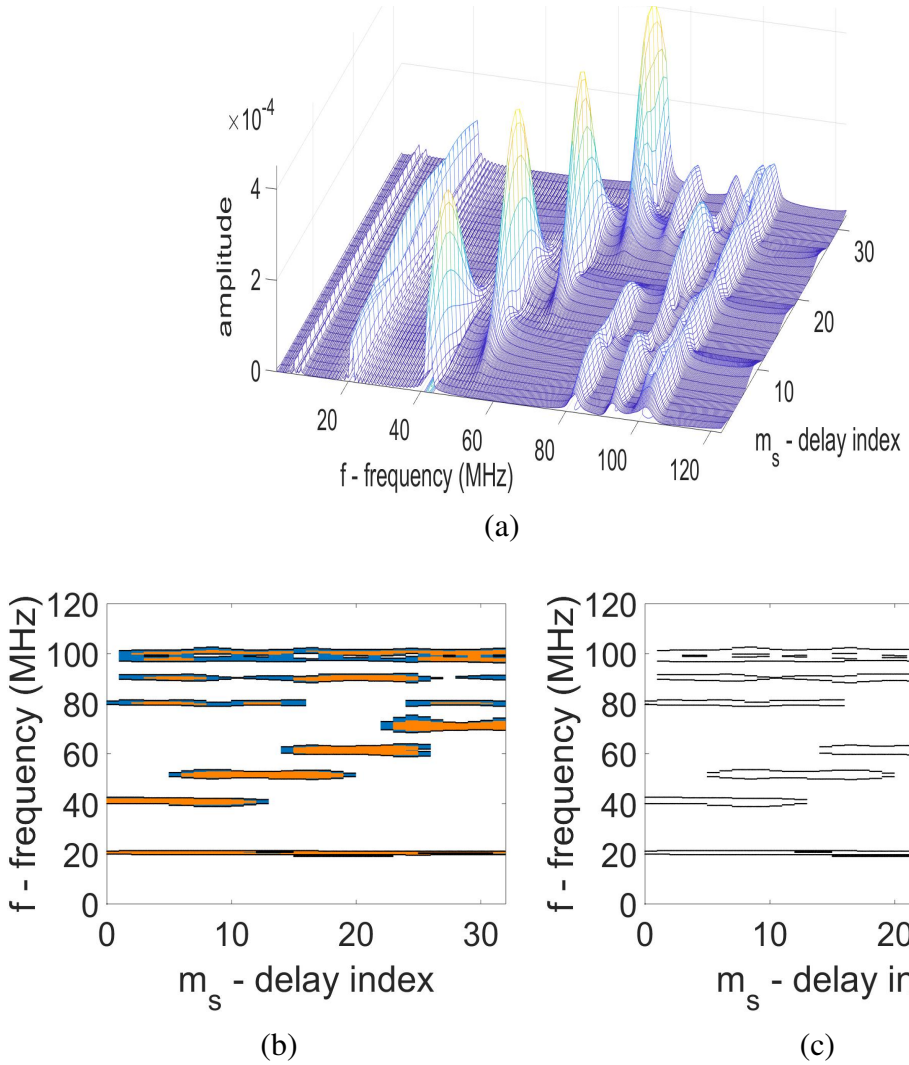


FIGURE 3.10: Signal detection and contours extracted from ST representation of the wideband signal with $P = 512$, $k = 17.5$ and $s = 32$: a) 3D representation; b) 2D representation of energy locations detected by threshold = $2\mu_{ST}$ (external dark-blue areas) and by threshold = $4\mu_{ST}$ (internal light-orange areas); c) contours of the energy locations with threshold = $2\mu_{ST}$. The SBW jumps among 41-51-61-71 MHz in the VHF band at different time instants

As shown in Fig. 3.10, signals can be detected by thresholding on the ST representation to obtain occupied areas in both the frequency and time domains. For the sake of clarity, the 3D representation corresponding to Fig. 3.9(c) is shown in this figure. These occupied regions correspond to locations where the wideband signal, such as the one represented in Fig. 3.10(a), is above a certain threshold which is set as k times the mean value of the ST representation computed on a 2-dimensional basis (k is a positive factor used to tune the threshold). In the example below, $k = \{2, 4\}$ and the threshold is denoted as μ_{ST}. In other words, let $\mathbf{X}$ the ST representation as in

Fig. 3.10(a). The threshold is computed as:

$$\text{threshold} = k \cdot \mu_{ST} = k \cdot \text{mean2}\,(\mathbf{X})$$

where mean2$(\cdot)$ is the mean function applied across 2 directions, namely time and frequency, and the outcome is a scalar value. Consequently, contours of the occupied regions for each signal, that are detected in the observed spectrum, can be extracted from the time-frequency representation of the wideband signal. An example is given as follow:

i. 3D representation with DR ($s = 32$) of the observed 0-120 MHz spectrum is in Fig. 3.10(a);

ii. energy locations obtained with threshold = $2\mu_{ST}$ and with threshold = $4\mu_{ST}$ are in Fig. 3.10(b) (external dark-blue areas and internal light-orange areas, respectively); μ_{ST} is the mean value of the ST representation with DR;

iii. contours of the occupied regions detected with threshold = $2\mu_{ST}$ are in Fig. 3.10(c).

In the middle of each figure, the non-stationary SBW signal can be easily seen which jumps in the VHF frequency range at frequencies 41-51-61-71 MHz where the signal is found at different time instants. This kind of analysis lays the foundations for extraction of dynamic features from the 2-dimensional contours of each moving signal such as bandwidth, central frequency, transmitting power or amplitude, and variance. This approach is applied to the proposed research discussed in Chapters 6 and 7 where, without loss of generality, the amplitude is used as feature.

A comparison with the STFT shows the grade of applicability to a high sampling rate framework and, in particular, to possible signal identification through features extracted from each of the signals detected in the time-frequency representation. Specifically, the obtained STFT representation is shown in Figs. 3.11(a)-3.11(d) for four different window lengths ($N = 128,\ 64,\ 32,\ 16$). A sensible choice for the length of the window, N, could reduce both the time resolution drop, happening when N is large, and the frequency resolution drop, happening when N is small. In any case, the STFT representations of wideband non-stationary signals are not sufficient for signal identification applications where high accuracy discriminative features are requested.

To show the effective benefits obtained with the proposed algorithm, Fig. 3.12 illustrates three different indicators: the time to create the $\mathbf{T}$ matrix, the time to compute the ST through Eq. (3.32), and the $L2$-norm error which illustrates the difference between the 2D TF representation without DR and the corresponding representation

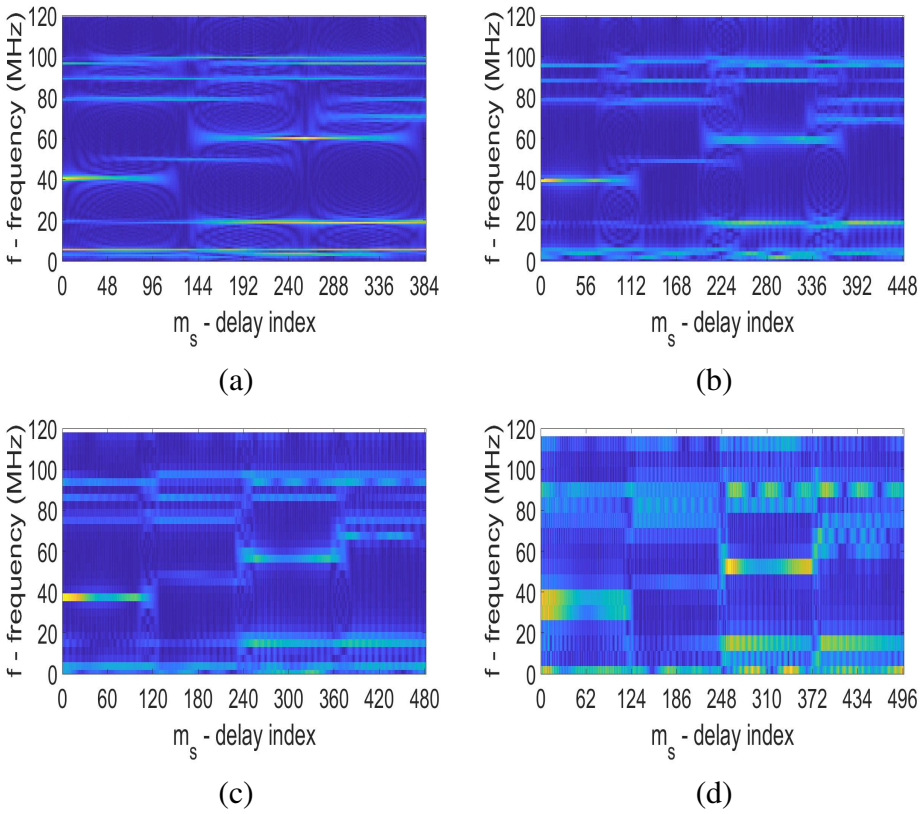


FIGURE 3.11: STFT representation of the wideband signal with $P = 512$. The length of the sliding window is: a) $N = 128$, b) $N = 64$, c) $N = 32$, and d) $N = 16$. The SBW jumps among 41-51-61-71 MHz at different time instants

with DR ($s = 64, 32, 16$). Specifically, let ϵ_n the normalized $L2$-norm error defined as $\frac{\|\mathbf{X}_s - \mathbf{X}_P\|_2}{\|\mathbf{X}_P\|_2}$, where the matrix $\mathbf{X}_s$ is the expanded ST representation with DR, obtained through P/s replicas of each column, the matrix $\mathbf{X}_P$ is the ST representation without DR, and $\|\cdot\|_2$ is the $L2$-norm function.

Figs. 3.12(a) and 3.12(b) show that in the conventional ST without DR the computational time to both generate the $\mathbf{T}$ matrix and perform the ST increases sharply when P is increased from 256 to 640. With $P = 1024$, Matlab is no longer capable of generating $\mathbf{T}$ because of its large size ($P \cdot P$ rows and P columns). While, the dual-resolution approach reduces dramatically the computational time and $P = 1024$ is also feasible. This means that the frequency resolution can be improved. Basically, the parameter k doesn't influence the computational time which is though increased by s.

Fig. 3.12(c), where $P = 512$, shows that both s and k influence ϵ_n. As expected, the error (mostly related to the time-resolution) increases when at least one between s and k decreases and it can be reduced by increasing k when s is small. In Fig. 3.12(c),

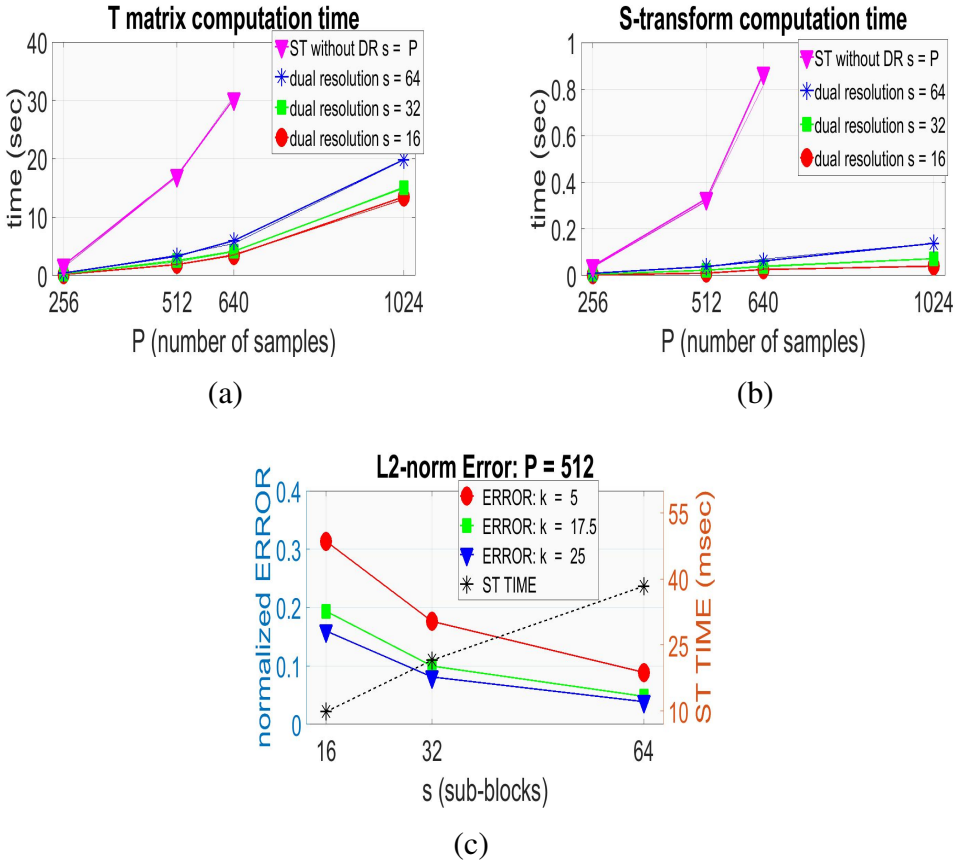


FIGURE 3.12: Performance indicators: a) time to generate **T**, b) ST computation time, c) normalized $L2$-norm error

the corresponding ST computation time is also included which is inversely proportional to the error. The same results hold when a different value for P is considered.

To know the percentage of improvement, both the computation time and the time to number of samples ratio for the proposed approach are compared with the corresponding values for the conventional ST. From Fig. 3.12(a), the '**T** matrix computation time' is reduced from 17.1 secs (ST without DR) to 1.9 (or 3.4) secs (with DR), when $P = 512$. This corresponds to

$$\frac{\Delta sec}{17.1}100\ (\%) = \frac{17.1-1.9\ (\text{or } 3.4)}{17.1}100\ (\%) = \frac{15.2\ (\text{or } 13.7)}{17.1}100\ (\%) = 88.89\ (\text{or } 80.12)\ \%.$$

In other words, when $P = 512$, just $\frac{1.9\ (\text{or } 3.4)}{17.1}100\ (\%) = 11.11\ (\text{or } 19.88)\ \%$ of the time necessary to generate the **T** matrix without DR is requested when DR is applied. In addition, 30.2 secs (ST without DR) and 3.5 (or 5.8) secs (with DR) have been measured when $P = 640$. This corresponds to

$$\frac{\Delta \text{sec}}{30.2}100\ (\%) = \frac{30.2-3.5\ (\text{or } 5.8)}{30.2}100\ (\%) = \frac{26.7\ (\text{or } 24.4)}{30.2}100\ (\%) = 88.41\ (\text{or } 80.79)\ \%.$$

Namely, when DR is applied, just $\frac{3.5 \text{ (or } 5.8)}{30.2}100$ (%) = 11.59 (or 19.21) % of the time necessary to generate the **T** matrix without DR is requested.

In the same picture, the time to number of samples ratio is $\frac{15}{1024} = 14.65 \cdot 10^{-3}$ for $P = 1024$ and $s = 32$ (with DR) and $\frac{30}{640} = 46.88 \cdot 10^{-3}$ for $P = 640$ and without DR; namely, more than 3 times smaller in ST with DR with respect to the conventional ST. Effectively, when DR is applied, the amount of samples to be ST-transformed can be doubled while guaranteeing comparable computational time to obtain the **T** matrix with conventional ST.

Similar conclusion results when values in Fig. 3.12(b) for 'ST computation time' are considered. Specifically, 0.33 secs (ST without DR) in comparison with 0.01 (or 0.04) secs (with DR), when $P = 512$, produces 3.03 (or 12.12) % which is the amount of necessary time to compute the ST with DR with respect the conventional ST. When $P = 640$, the measured time is 0.87 secs (ST without DR) and 0.025 (or 0.06) secs (with DR), corresponding to 2.87 % and 6.90 %, respectively.

In this case, the time to number of samples ratio is $\frac{0.07}{1024} = 68.36 \cdot 10^{-6}$ for $P = 1024$ and $s = 32$ (with DR) and $\frac{0.87}{640} = 1.36 \cdot 10^{-3}$ for $P = 640$ and without DR; namely, about 20 times smaller in ST with DR with respect to the conventional ST. Effectively, when DR is applied, the number of samples that guarantees comparable ST computational time, with respect to the case without DR, is much larger; till 4 times larger, when $s = 32$.

3.5.5 Conclusion and Future Directions

In many applications, such as dynamic spectrum sensing and dynamic spectrum access, processing wideband signals sampled at high sampling rate is the main objective. Both time and frequency information should be contemporary processed to represent the dynamic nature of the spectrum.

The novel ST-based dual-resolution approach proposed in this work overcomes the limitations of the conventional ST when applied to wideband signals by increasing the time delay of its sliding window and allows to achieve a good trade-off between time-resolution and computational time which can now be regulated in real-time applications. Applied to real data, the ST with DR has shown a significant reduction of the computational time with respect to the case without DR, both to generate the matrix **T** and to calculate the ST, with just a slight decrease in time-resolution and TF representation accuracy. In particular, just 11% to 20% of the time necessary to generate the **T** matrix without dual-resolution is requested and 3% to 12% of the ST computational time with respect to the conventional ST. In addition, discrete series

with more samples, which usually happens with wideband signals, can be also processed. Indeed, the number of samples can be till 4 times larger while guaranteeing comparable computational time.

A comparison between ST and STFT time-frequency representations shows that the STFT representations of wideband non-stationary signals are not sufficient to guarantee reliability in applications, like signal identification, where high accuracy discriminative features extracted from each of the signals are requested. In this work, the concept of reliability is mainly related to features that can be extracted from TF representation; at this stage, a qualitative definition of this concept is considered. Consequently, from representations in Fig. 3.11, it results in a reduced quality of the images and, then, of the features that can be extracted, at least in one of the two domains (time/frequency). In future work, where extracted features are effectively employed, the degree of achieved reliability will be assessed by quantifying the concept through metrics related to the implemented algorithm such as accuracy in the classification rate when, for example, a classifier is applied to recognize the different signals inside the spectrum of interest. In signal identification, the rate of identification and its accuracy (based on the extracted features) is also a quantitative approach for the concept of reliability.

Moreover, the length of the sliding window in STFT should be chosen very accurately, mostly due to the time-frequency resolution trade-off. Since the proposed approach is meant to be applied in a context where information about dynamics of signals should be extracted from the TF representation, optimization of rate of detection of jumps or actions, rate of correct (re-)identification of signals when they appear somewhere in the spectrum, or accuracy of trajectories would be possible metrics that allow us to set a suitable length of the sliding window of the STFT. Results from the comparison made ST as the preferred choice for developing future work based on modelling the dynamics of signals.

Future work also assumes the presence of non-stationarities within sub-blocks and detects them to reduce the basic assumption of the proposed approach, and combines the DR to Discrete Orthogonal S-Transform (DOST) for a more efficient representation of ST [48]. In addition, dynamic models and learning mechanisms, based on ST computational gain and precision of detecting hop frequencies, can be implemented as part of Learning Dynamic Jamming models toward cognitive dynamic systems (CDSs) for PHY-layer security and Cognitive Radio.

Specifically, both time and frequency information are obtained from dynamic signals with controllable resolution in both the domains. The corresponding contours are then extracted. Future directions in this framework could include application of techniques and algorithms used for data analysis, image and video processing, robotics,

and so forth, to wireless communications. Indeed, general concepts such as entity, state, and trajectory can be also specified in CR as wireless signal, central frequency-bandwidth-amplitude-variance, and time-frequency information, respectively. Considering that, dynamic models are learned from the observed features to represent moving signals through probabilistic distributions.

Validation of the ST-based DR will also be performed on TVWS signals in the 470-790 MHz band in the UK.

Bibliography

[1] M. I. M. Alfaqawi, J. Chebil, M. H. Habaebi, and D. Datla. Wireless distributed computing for cyclostationary feature detection. *Digital Communications and Networks*, 2(1):47–56, 2016.

[2] A. Ali and W. Hamouda. Advances on spectrum sensing for cognitive radio networks: Theory and applications. *IEEE Communications Surveys Tutorials*, 19(2):1277–1304, Secondquarter 2017.

[3] J. B. Allen and L. R. Rabiner. A unified approach to short-time fourier analysis and synthesis. *Proceedings of the IEEE*, 65(11):1558–1564, November 1977.

[4] E. April. Ad-a258 307 the advantage of cyclic spectral analysis (u). In *technical note,Defence Research Establishment Ottawa*, October 1991.

[5] Y. Baiqiang and H. Yigang. A fast matrix inverse s-transform algorithm for mcg denoise. In *IEEE International Conference on Electronic Measurement and Instruments (ICEMI)*, pages 315–319, July 2015.

[6] S. V. Bhagate and S. Patil. Maximizing spectrum utilization in cognitive radio network. In *2017 International Conference on Big Data, IoT and Data Science (BID)*, pages 82–90, Dec 2017.

[7] M. Born and P. Jordan. Zur quantenmechanik. *Zeitschrift für Physik*, 34(1):858–888, Dec 1925.

[8] B. Bouachache and P. Flandrin. Wigner-ville analysis of time-varying signals. In *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP, Paris, France*, pages 1329 – 1332, May 1982.

[9] R.A. Brown, M.L. Lauzon, and R. Frayne. A general description of linear time-frequency transforms and formulation of a fast, invertible transform that samples

the continuous s-transform spectrum nonredundantly. *IEEE Transactions on Signal Processing*, 58(1):281–290, January 2010.

[10] G. Y. Chang, S. Y. Wang, and Y. X. Liu. A jamming-resistant channel hopping scheme for cognitive radio networks. *IEEE Transactions on Wireless Communications*, 16(10):6712–6725, Oct 2017.

[11] W.L. Chin, J.M. Li, and H.H. Chen. Low-complexity energy detection for spectrum sensing with random arrivals of primary users. *IEEE Transactions on Vehicular Technology*, 65(2):947–952, February 2015.

[12] H. I. Choi and W. J. Williams. Improved time-frequency representation of multicomponent signals using exponential kernels. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(6):862–871, June 1989.

[13] L. Cohen. Time-frequency distributions-a review. *Proceedings of the IEEE*, 77(7):941–981, July 1989.

[14] K. Dabčević, L. Marcenaro, and C.S. Regazzoni. Security in cognitive radio networks. In *Evolution of Cognitive Networks and Self-Adaptive Communication Systems*, pages 301–335, IGI Global, 2013.

[15] K. Dabčević, M.O. Mughal, L. Marcenaro, and C. S. Regazzoni. Spectrum intelligence for interference mitigation for cognitive radio terminals. In *Wireless Innovation Forum European Conference on Communications Technologies and Software Defined Radio (WInnComm- Europe), Rome, Italy*, November 2014.

[16] K. Dabčević, M.O. Mughal, L. Marcenaro, and C. S. Regazzoni. Cognitive radio as the facilitator for advanced communications electronic warfare solutions. *Journal of Signal Processing Systems*, 83(1):29–44, April 2016.

[17] I. Daubechies. The wavelet transform, time-frequency localization and signal analysis. *IEEE Transactions on Information Theory*, 36(5):961–1005, September 1990.

[18] F. F. Digham, M. Alouini, and M. K. Simon. On the energy detection of unknown signals over fading channels. *IEEE Transactions on Communications*, 55(1):21–24, Jan 2007.

[19] S. Dzakmic, T. Namas, and I. Dzafic. Fault classification using multi-resolution analysis and discrete wavelet transforms. In *2017 XXVI International Conference on Information, Communication and Automation Technologies (ICAT), Sarajevo, Bosnia and Herzegovina*, pages 1–6, October 2017.

[20] W. Gardner. Spectral correlation of modulated signals: Part i - analog modulation. *IEEE Transactions on Communications*, 35(6):584–594, June 1987.

[21] W. Gardner, W. Brown, and Chih-Kang Chen. Spectral correlation of modulated signals: Part ii - digital modulation. *IEEE Transactions on Communications*, 35(6):595–601, June 1987.

[22] W. Gardner and J.A. Cadzow. Statistical spectral analysis: a nonprobabilistic theory. *Applied Optics*, 29:1399, 1990.

[23] W. Gardner and L. Franks. Characterization of cyclostationary random signal processes. *IEEE Transactions on Information Theory*, 21(1):4–14, January 1975.

[24] W. A. Gardner. *Statistical spectral analysis: A nonprobabilistic theory*. Englewood Cliffs, N.J: Prentice Hall, 1988.

[25] W. A. Gardner. Exploitation of spectral redundancy in cyclostationary signals. *IEEE Signal Processing Magazine*, 8(2):14–36, April 1991.

[26] S. Haykin, D. J. Thomson, , and J. H. Reed. Spectrum sensing for cognitive radio. *Proceedings of the IEEE*, 97(5):849–877, April 2009.

[27] N. E. Huang and *et al*.. The empirical mode decomposition and hilbert spectrum for nonlinear and nonstationary time series analysis. *Proc. Roy. Soc. London Ser. A, Math., Phys. Eng. Sci.*, 454(1971):903–995, 1998.

[28] M. A. De Jesus, M. Teixeira, L. Vicente, and Y. Rodriguez. Nonuniform discrete short-time fourier transform a goertzel filter bank versus a fir filtering approach. In *2006 49th IEEE International Midwest Symposium on Circuits and Systems*, volume 2, pages 188–192, August 2006.

[29] B. Jokanovic, M. G. Amin, and F. Ahmad. Effect of data representations on deep learning in fall detection. In *2016 IEEE Sensor Array and Multichannel Signal Processing Workshop (SAM)*, pages 1–5, July 2016.

[30] T. Kailath and H. V. Poor. Detection of stochastic processes. *IEEE Transactions on Information Theory*, 44(6):2230–2259, Oct 1998.

[31] S. Kapoor, S. Rao, and G. Singh. Opportunistic spectrum sensing by employing matched filter in cognitive radio network. In *2011 International Conference on Communication Systems and Network Technologies*, pages 580–583, June 2011.

[32] K. Konopko. An implementation of the cohen's class time-frequency distributions on a massively parallel processor. 2012.

[33] S. Kozłowski. Implementation and verification of cyclostationary feature detector for dvb-t signals. *IET Signal Processing*, 10(2):162–167, March 2016.

[34] E. Like, V. Chakravarthy, R. Husnay, and Z. Wu. Modulation recognition in multipath fading channels using cyclic spectral analysis. In *IEEE Global Telecommunications Conference (GLOBECOM)*, pages 1–6, November 2008.

[35] J. Lunden, V. Koivunen, and H. V. Poor. Spectrum exploration and exploitation for cognitive radio: Recent advances. *IEEE Signal Processing Magazine*, 32(3):123–140, May 2015.

[36] G. Malegori and G. Ferrini. Tip-sample interactions on graphite studied using the wavelet transform. *Beilstein Journal of Nanotechnology*, 1:172–181, December 2010.

[37] J. Mohammadi, S. Stańczak, and M. Zheng. Joint spectrum sensing and jamming detection with correlated channels in cognitive radio networks. In *2015 IEEE International Conference on Communication Workshop (ICCW)*, pages 889–894, June 2015.

[38] A. Nafkha, B. Aziz, M. Naoues, and A. Kliks. Cyclostationarity-based versus eigenvalues-based algorithms for spectrum sensing in cognitive radio systems: Experimental evaluation using gnu radio and usrp. In *2015 IEEE 11th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, pages 310–315, Oct 2015.

[39] A. Nafkha, M. Naoues, K. Cichon, and A. Kliks. Experimental spectrum sensing measurements using usrp software radio platform and gnu-radio. In *2014 9th International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CROWNCOM)*, pages 429–434, June 2014.

[40] T. Nawaz, M.O. Mughal, L. Marcenaro, and C.S. Regazzoni. Exploiting cyclic features for jammer detection in wide-band cognitive radios. In *WInnComm-Europe' 15, Erlangen, Germany*, October 2015.

[41] S. Okamura. The short time fourier transform and local signals. In *Dissertations, 58. Department of Statistics, Carnegie Mellon University Pittsburgh, Pennsylvania*, June 2011.

[42] K. Patel, N. C. Kurian, and N. V. George. Time frequency analysis: A sparse s transform approach. In *2016 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS)*, pages 1–4, October 2016.

[43] V. R. Petty, R. Rajbanshi, D. Datla, F. Weidling, D. DePardo, P. J. Kolodzy, M. J. Marcus, A. M. Wyglinski, J. B. Evans, G. J. Minden, and J. A. Roberts. Feasibility of dynamic spectrum access in underutilized television bands. In *2007 2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks*, pages 331–339, April 2007.

[44] R.S. Roberts, W.A. Brown, and H.H. Loomis. Computationally efficient algorithms for cyclic spectral analysis. *IEEE Signal Processing Magazine*, 8(2):38–49, April 1991.

[45] U. Satija, M. Mohanty, and B. Ramkumar. Automatic modulation classification using s-transform based features. In *2nd International Conference on Signal Processing and Integrated Networks (ICSPIN)*, pages 708–712, February 2015.

[46] C. Simon, S. Ventosa, M. Schimmel, A. Heldring, J.J. Da nobeitia, J. Gallart, and A. Mànuel. The s-transform and its inverses: Side effects of discretizing and filtering. *IEEE Transactions on Signal Processing*, 55(10):4928–4937, October 2007.

[47] R. G. Stockwell, L. Mansinha, and R. P. Lowe. Localization of the complex spectrum: the s transform. *IEEE Transactions on Signal Processing*, 44(4):998–1001, April 1996.

[48] R.G. Stockwell. A basis for efficient representation of the s-transform. *Digital Signal Processing*, 17(1):371 – 393, January 2007.

[49] Z. Tian and G.B. Giannakis. A wavelet approach to wideband spectrum sensing for cognitive radios. In *IEEE 1st International Conference on Cognitive Radio Oriented Wireless Networks and Communications, CROWNCOM*, pages 1 – 5, July 2006.

[50] H. Urkowitz. Energy detection of unknown deterministic signals. *Proceedings of the IEEE*, 55(4):523–531, April 1967.

[51] P. Wang, J. Fang, N. Han, and H. Li. Multiantenna-assisted spectrum sensing for cognitive radio. *IEEE Transactions on Vehicular Technology*, 59(4):1791–1800, May 2010.

[52] R. Wang and M. Tao. Blind spectrum sensing by information theoretic criteria for cognitive radios. *IEEE Transactions on Vehicular Technology*, 59(8):3806–3817, Oct 2010.

[53] Z. Wu, E. Like, and V. Chakravarthy. Reliable modulation classification at low snr using spectral correlation. In *4th IEEE Consumer Communications and Networking Conference*, pages 1134–1138, January 2007.

[54] Z. Xinzhi, G. Feifei, C. Rong, and J. Tao. Matched filter based spectrum sensing when primary user has multiple power levels. *China Communications*, 12(2):21–31, April 2015.

[55] H. Yousry1, F. Newagy, S. ElRamly, and M. Ibrahim. Wireless microphone sensing using cyclostationary detector. In *2012 Innovations on Communication Theory (INCT2012)*, 01 2012.

[56] T. Yucek and H. Arslan. A survey of spectrum sensing algorithms for cognitive radio applications. *IEEE Communications Surveys Tutorials*, 11(1):116–130, March 2009.

[57] B. Zayen, A. Hayar, and K. Kansanen. Blind spectrum sensing for cognitive radio based on signal space dimension estimation. In *2009 IEEE International Conference on Communications (ICC)*, pages 1–5, June 2009.

[58] Y. Zeng, C. L. Koh, and Y. C. Liang. Maximum eigenvalue detection: Theory and application. *2008 IEEE International Conference on Communications (ICC)*, pages 4160–4164, 2008.

[59] R. Zhang, T. J. Lim, Y. Liang, and Y. Zeng. Multi-antenna based spectrum sensing for cognitive radios: A glrt approach. *IEEE Transactions on Communications*, 58(1):84–88, January 2010.

[60] Y. Zhao, Y. Wu, J. Wang, X. Zhong, and L. Mei. Wavelet transform for spectrum sensing in cognitive radio networks. In *International Conference on Audio, Language and Image Processing (ICALIP), Shanghai, China*, pages 565–569, June 2014.

4 Learning Dynamic Models from spectrum representation *

A general framework for the learning phase, with some theoretical descriptions and practical applications, is now described in this chapter. This corresponds to the machine learning block **(stage 2)** in Fig. 4.1 of the general cognitive capability diagram where statistical representation of the signals in the spectrum of interest can be learnt. Two main directions are based on Artificial Neural Networks and Bayesian Networks, respectively.

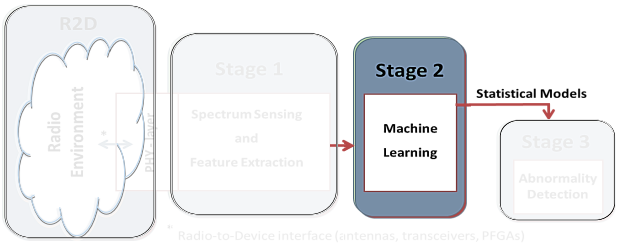


FIGURE 4.1: Learning phase (stage 2)

4.1 Motivation

An *Interactive and Cognitive Environment (ICE)* [9] can be defined as a physical environment with artificially extended capabilities obtained through digital artificial cognition based on Information and Communication Technologies (ICT). An ICE requires the capability to understand and to actively modulate human-machine interactions by learning from experiences.

A CR can be seen as an actor of an ICE. The physical nature of the appropriate environment for a CR is the electromagnetic (EM) spectrum. A CR can perform a cognitive cycle by sending signals on the spectrum by means of antennas and can sense the same spectrum always by using antennas. A CR can be become progressively smarter if it can learn from experiences models that associate signal it can sense to signals it can send, i.e. if it can learn models able to adapt the cognitive cycles it has to apply to be maintained in dynamic equilibrium with the external environment. Recent advances in both SDR and ML can make this possible. SDR is a paradigm

* Work published (see [2] in *List of publications and under review work*)

which enables a full software implementation of radio devices making them general purpose and dynamically reconfigurable devices (even in on-line applications). This means that internal parameters can be regulated and controlled by software according to some internal and/or external state. Implementation of real-time adaptive systems is then feasible. By employing ML, a paradigm is available consisting of methodologies and techniques developed to translate big amounts of sensed data into classification and generative models that can be used to detect and predict situations. Self-awareness can be introduced in existing systems (such as radar, robots, and wireless equipments), that can enable radios to become more adaptive, cognitive, and interactive.

The success encountered in this field has resulted in innovative techniques in Radio Communications like *Dynamic Spectrum Access* (DSA) which enables opportunistic transmission on shared spectrum and *defence against jamming attacks* to address the physical layer (PHY-layer) security problem. *Spectrum Intelligence (SI)* and *TVWS* are two current applications of Interactive and Cognitive devices as described in this chapter.

Since the environment is assumed to be dynamic, *time-information* is a decisive factor when analysing and processing signals. To this end, *time-frequency analysis* is the tool which retains both frequency and time information of signals, namely not only where signals are inside the spectrum of interest but also when they are in specific bands that represent the contextual spectrum environment. Specifically, dynamic features can be extracted from the 2-dimensional representations (such as bandwidth, central frequency, transmitting power, and shape) of each signal. These features constitute a way to indirectly observe the state (hidden or non-observable) in the dynamical model for each entity. The spectrum of interest is the dynamic environment while signals inside it represents the observations by which the hidden state of the entities can be measured. Once the problem is described, signal representation is the framework which enables entities and interactions modelling. There are several techniques to perform signal representation, namely modelling entities in the specific environment. Basically, dynamic systems are used in which the state is observed through noisy measurements. They consist of both a *dynamic model* of the state and a *measurement model*. Transition probabilities of the state refers to the probabilities of changing from an initial state to a new state. In an ICE scenario, single entities are described by stand-alone models while interacting (multiple) entities are described by linked coupled models.

In this chapter, different features and signal representations are presented along with probabilistic models. This is necessary to go through the learning process in

order to estimate and predict state and transitions for each single entity as well as interactions for multiple interacting entities.

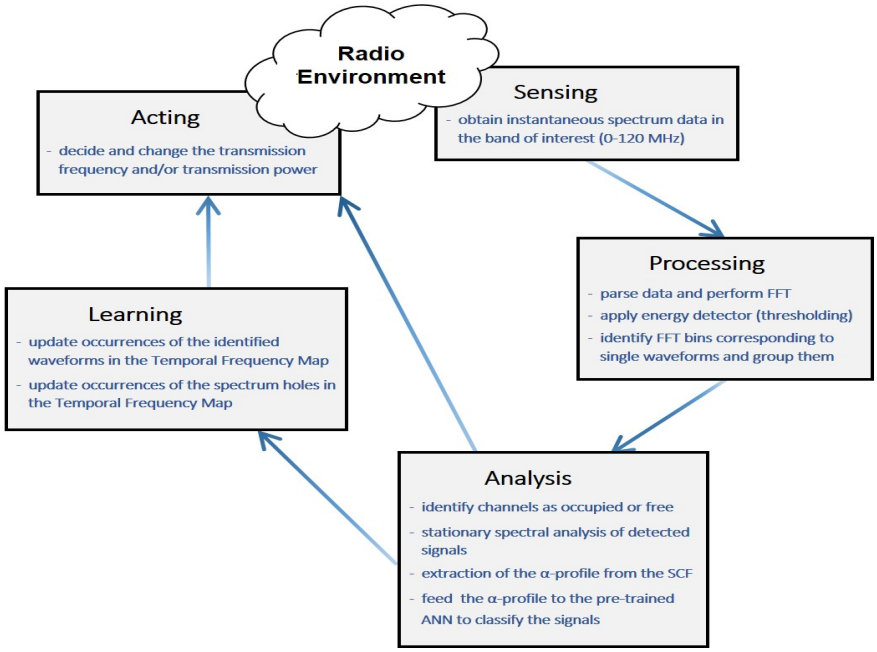


FIGURE 4.2: Proposed artificial intelligence enabled (AI-enabled) cognitive radio framework with CSI algorithm

4.2 Example (cont'd): Spectrum correlation in modulated signals *

Before going through learning dynamic models, let us continue the example of interference mitigation in wideband radios discussed in Sec. 3.3. By using spectrum correlation for modulation recognition, a Cyclic Spectrum Intelligence algorithm based on α-profile is proposed in the AI-enabled cognitive radio framework as an alternative to the existing solutions in [5, 6]. This is an example about learning static models. The algorithm is based on a cognitive cycle, shown in Fig. 4.2, to acquire self-awareness from spectrum measurements by differentiating friendly waveforms from potential malicious signals according to their modulation scheme. Firstly, instantaneous spectrum data is collected at sensing stage through a SDR testbed and, then, the wideband spectrum is processed to identify fast Fourier transform (FFT) bins corresponding to single waveforms and group them. Afterwards, analysis is applied. Unlike the spectrum intelligence in [5, 6], CSI employs a CFD to extract more advanced features and

* Work published (see [3] in *List of publications and under review work*)

an Artificial Neural Network (ANN) [19] to classify the observed signals and perform interference mitigation in a WB spectrum. Specifically, the cyclostationary feature, called α-profile, of each detected signal in the band-of-interest is classified through an ANN-based recognizer. Results are promising and show high classification rate even for low transmission power case. Moreover, CSI produces better or comparable classification rates with respect to existing approaches. Cyclic spectral analysis and complexity are discussed in Secs. 3.3.2 and 4.2.1. For the sake of completeness, learning and acting blocks are also introduced.

The main novelty is then the newly proposed analysis block as part of a new CSI algorithm for interference mitigation in wideband radios at system level. This improves the resulting performance of two major previous works in the literature for spectrum intelligence algorithms. Moreover, unlike previous work, the proposed classification scheme is tested in a complex dynamic environment which is the most probable scenario in CR applications.

In the following, the proposed CSI algorithm is described in Sec. 4.2.1 along with the neural network classifier, and the corresponding complexity analysis. While, in Sec. 4.2.2, *Learning* and *Acting* blocks conclude the cognitive cycle. Validation of the proposed approach with experimental data is analysed in Sec. 4.2.3 along with a comparison with two previous works. Some conclusive considerations and future work are also given.

4.2.1 Cyclic Spectrum Intelligence (CSI) algorithm

The principal idea behind the CSI algorithm is to continuously monitor relevant radio-frequency (RF) spectrum activities, identify potential threats to communication, and take proactive measures to ensure communication robustness and secrecy. For doing so, the algorithm relies on the reliable spectrum sensing mechanism, correct identification and extraction of the relevant parameters, and secure software unsubjected to tampering. In comparison with the spectrum intelligence algorithm described in [5], the proposed CSI algorithm employs a cyclostationary feature algorithm to extract the α-profile feature from the detected signals which is then fed to a neural network to classify the waveforms present into the observed spectrum.

The functional process of the CSI algorithm can be represented in the form of the Cognitive Cycle, as shown in Fig. 4.2, consisting of 5 blocks: *Sensing*, *Processing*, *Analysis* (described in Secs. 3.3.1 and 3.3.2), *Learning*, and *Acting*. *Sensing* and *Acting* blocks represent the interface with the external radio environment. The remaining blocks in the Cognitive Cycle and validation of the spectral correlation-based detector with a neural network classifier are now described.

Neural network classifier: the proposed system uses an ANN as a classifier due to its ease of implementation and potential to generalize any carrier frequency, symbol rate and phase offset. The system was designed to classify PM (potential malicious) and FR (friendly) signals. The ANN is trained to identify these two classes of signals (PM and FR). The SCF of the detected signals produces a large amount of data, which makes impossible for a classifier to work on it in real time. In order to reduce the amount of data for a classification stage, the α-profile is used as input feature for an ANN. Accordingly, the proposed ANN in Fig. 4.3 is composed of I inputs related to the dimensionality of α-profile, a single hidden layer whose N neurons use the hyperbolic tangent sigmoid as neural transfer function, and an output layer of two neurons related to each class of signal considered in this work. Each output value is in the range $(0, 1)$. Accordingly, the output class with the highest value between $(0, 1)$ is considered as the signal class. An ANN training based on the scaled conjugate gradient (SCG) back-propagation [18] is adopted.

The selection of a single hidden layer is based on the classification process simplicity of this particular problem, it was found that with a single layer results over the 99% of true positive classification were obtained for the 2 types of signal classes considered in this work. Employing more hidden layers would increment the training time and overall results would not be significantly improved. The corresponding pseudo-code of the proposed algorithm is outlined in Algorithm 1.

It can be summarized as follows: the receiver observes a WB signal and then energy detection and pre-processing are performed. The α-profile of SCF for each detected sub-signal is subsequently extracted. After that, detected signals go through

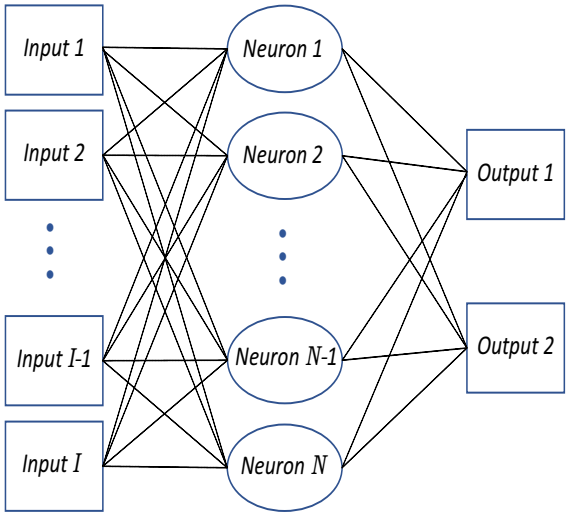


FIGURE 4.3: Proposed Artificial Neural Network used as classifier in the CSI algorithm with I inputs, one hidden layer with N neurons, and 2 outputs

Algorithm 1 Pseudo-code for proposed algorithm

function SIGNAL RECOGNITION IN WB SPECTRUM
  Set the number of bursts to be acquired $\rightarrow k$
  Receive and sample the wideband signal at or above Nyquist rate for all k bursts $\rightarrow N_S$ amplitude values
  Data parsing $\rightarrow N_S = 2^x$ amplitude values
  Perform FFT $\rightarrow \frac{N_S}{2}$ frequency bins with magnitudes M
  Fisrt smooth by moving average
  Calculate mean value of $M \rightarrow M_{mean}$
  Based on M_{mean}, set the energy thethreshold $\rightarrow \hat{\eta}$
  for $i = 1$ to $\frac{N_S}{2}$ **do** (for each frequency bin)
    if $M(i) > \hat{\eta}$ **then**
      Bin i belongs to the signal
      Change channel state of bin i to "occupied"
      if any of $M(i-K) : M(i-1) > M_T$ **then**
        Group these bins as a single waveform
        Perform waveform smoothing
      end if
    end if
  end for
  Estimate the SCF of detected signals
  Extract the α-profile
  Feed α-profile to previously trained Neural Network
  Decision $\leftarrow$ Licit or Jammer
end function

the classification process. The α-profile of detected signals are fed to a previously trained ANN. The ANN classifies the received narrowband signal, in the WB spectrum, as either a licit or a potential malicious user.

Complexity analysis (ANN computation complexity): a feedforward neural network is characterised by the total number of weights and biases, N, which form a weight vector $\mathbf{w}$. In the conventional back-propagation (BP) algorithm, a global error function $E(\mathbf{w})$ can be computed with one forward step, while its gradient $E'(\mathbf{w})$ with one forward and one backward step. The complexity of calculating the error function and its gradient is $O(N^2)$ and $O(3N^2)$, respectively. The BP algorithm often behaves very badly on large-scale problems and user dependent parameters, like learning rate, determine its performance.

Alternatively, conjugate direction methods, such as the conjugate gradients (CG) method, exploit second order approximation of the error function which, in addition to the gradient, utilizes the Hessian matrix denoted as $E''(\mathbf{w})$. There exists a unique global minimum only if the Hessian matrix is positive definite. However, in many cases, it has shown to be indefinite in different areas of the weight space and the CG

fails in the attempt to minimize the error function. Further details can be found in [18] which also proposes the scaled conjugate gradient back-propagation algorithm. As mentioned in the *Neural network classifier* paragraph, the SCG is adopted to train the ANN used as classifier. In this case, a calculation complexity per iteration of $O(7N^2)$ has been determined. When the algorithm is implemented, this complexity can be reduced to $O(6N^2)$.

4.2.2 Learning and acting algorithms

After the new *Analysis* block proposed in this work and belonging to the cognitive cycle, *Leaning* and *Acting* blocks presented in [6] conclude the proposed CSI algorithm.

Learning: after having identified occupied channels and spectrum holes in the 0-120 MHz band, and classified the detected signals through the cyclostationary feature algorithm with an artificial neural network as classifier, the CSI algorithm is thought to include a *learning* process strategy based on a Temporal Frequency Map which summarises previous occurrences of FR waveforms, PM waveforms, and spectrum holes for each channel inside the band-of-interest.

Acting: finally, based on the processed spectrum information, current transmission parameters (e.g. channel and transmission power) and the history obtained from the Temporal Frequency Map, the CR device may decide to *act* in order to improve its chances of reliable transmission. The actions consist of proactive changes of the transmission frequency (channel surfing), or the transmission power whenever a threat is detected, namely, when a PM waveform is identified on a channel close to the channel currently used for transmission. Highest priority is given to actions which avoid channels with history of occurrences of PM waveforms, followed by the channels with history of occurrences of FR waveforms.

Further applications include the capability of the cognitive system to learn from the actions of a human operator through a graphical user interface (GUI) allowing the human operator to overrule the decision of the cognitive algorithm and change transmission parameters such as the operating frequency and the transmission power. The role of the GUI is then to allow the human operator to take decisions irrespectively of the decisions of the CSI algorithm. However, it also presents an interesting motivation for considering principles of cognitive refinement, i.e., refining the reasoning behaviour of the algorithm, which is currently policy-based, by learning from the actions of the human operator.

4.2.3 Validation of the proposed algorithm

In order to evaluate the performance of the newly introduced *analysis* block of the cognitive cycle in the CSI algorithm (based on spectral correlation detector and neural network classifier for interference mitigation) a set of experiments is performed using the software defined radio testbed architecture described in Sec. 4.2.1. The sampling rate is set at Nyquist rate for each type of detected signals. In the experiments, the SBW signal represents the "potentially malicious" waveform and is transmitted by the transmitting HH. Its transmission parameters are given different values according to Tab. 3.1 in Sec. 3.3.1, namely the carrier frequency and the transmission power of the SBW signal assume one among the nine configurations with full, half, and one-tenth transmission power. All other detected signals mentioned in Sec. 3.3.1 are considered as "friendly". The objective of this section is to analyse the performance of the proposed algorithm in classifying the signals detected in the 0-120 MHz band-of-interest based on the α-profile which is extracted from each detected waveform in the wideband spectrum and then fed to the neural network used as classifier.

As introduced in Sec. 3.3.3, each generated α-profile consists of 200 cyclic frequency points which are the input of the ANN ($I = 200$ in Fig. 4.3). In this section, it is shown that the ANN is effectively able to classify the signal independently of the part of the spectrum in which it is detected. The ANN architecture employed to validate the *analysis* block of the CSI algorithm consists of 10 neurons in the hidden layer ($N = 10$ in Fig. 4.3). For the experiments, a dataset composed of 4.500 samples (500 for each configurations) is used in order to train (70%), validate (15%) and test (15%) the ANN architecture. Testing of the ANN showed a classification rate approximately equal to 1.

After having trained and tested the ANN with 4500 samples, further assessment of the performance is obtained with a different set of waveforms which are tested by using the trained neural network. Specifically, the performance of the system is evaluated for 1000 independent testing samples for each of the configuration with different carrier frequencies (51-61-71 MHz) and transmission powers (7, 4, and -3 dBm). The confusion matrices in Tab. 4.1 show the classification accuracy for the 9 different configurations. The proposed method based on the cyclostationary feature algorithm combined with an artificial neural network provides good performance with high classification rate in most of the configurations, even for low transmission power case.

A comparison with results in the literature is also presented. For example, in [5] spectrum sensing is performed through an energy detector to identify the occupied bands in the widedand spectrum. Center frequency, bandwidth and maximum value of magnitude are extracted for each of the identified narrowband waveforms. To

FULL POWER (7 dBm)

SIGNAL \ CLASS	FR	PM	FR	PM	FR	PM
FR	960	40	970	30	981	19
PM	0	1000	1	999	0	1000
	51MHz (I)		*61MHz (II)*		*71MHz (III)*	

HALF POWER (4 dBm)

SIGNAL \ CLASS	FR	PM	FR	PM	FR	PM
FR	985	15	996	4	999	1
PM	0	1000	0	1000	0	1000
	51MHz (IV)		*61MHz (V)*		*71MHz (VI)*	

ONETENTH POWER (-3 dBm)

SIGNAL \ CLASS	FR	PM	FR	PM	FR	PM
FR	997	3	997	3	997	3
PM	20	980	88	912	0	1000
	51MHz (VII)		*61MHz (VIII)*		*71MHz (IX)*	

TABLE 4.1: Confusion matrices with absolute values from testing on independent samples applied to the neural network; each configuration (from *I* to *IX*) has been tested separately

perform classification, these parameters extracted from the identified waveforms are compared to parameters stored in a dataset containing pre-defined parameters of the FR and/or PM waveforms. This results in classification of each waveform as either FR or PM.

For the sake of clarify, comparison is made on the percentage values in the confusion matrices.

In Tab. 4.2, the percentage values relative to all the configurations analysed in this work, from *I* to *IX*, are shown, while in Tab. 4.3 results from [5] can be seen where observed wideband spectrum, corresponding to both 61 MHz and 71 MHz centre frequencies, have been contemporary tested with 200 independent samples (or bursts). Two different feature vectors have been investigated, namely bandwidths of the detected waveforms in Tab. 4.3(a), while both bandwidths and magnitudes in Tab. 4.3(b). Specifically, different confusion matrices are obtained for each of the two type of feature vector by varying the number of bursts (N_b) - over three levels (1, 3, and 5) - which are averaged in order to increase the frequency resolution of the observed spectrum and to investigate its influence to the classification accuracy.

FULL POWER (7 dBm)

SIGNAL \ CLASS	FR	PM	FR	PM	FR	PM
FR	96.0%	4.0%	97.0%	3.0%	98.1%	1.9%
PM	0%	100%	0.1%	99.9%	0%	100%
	51MHz (I)		*61MHz (II)*		*71MHz (III)*	

HALF POWER (4 dBm)

SIGNAL \ CLASS	FR	PM	FR	PM	FR	PM
FR	98.5%	1.5%	99.6%	0.4%	99.9%	0.1%
PM	0%	100%	0%	100%	0%	100%
	51MHz (IV)		*61MHz (V)*		*71MHz (VI)*	

ONETENTH POWER (-3 dBm)

SIGNAL \ CLASS	FR	PM	FR	PM	FR	PM
FR	99.7%	0.3%	99.7%	0.3%	99.7%	0.3%
PM	2.0%	98.0%	8.8%	91.2%	0%	100%
	51MHz (VII)		*61MHz (VIII)*		*71MHz (IX)*	

TABLE 4.2: Confusion matrices with percentage values corresponding to the absolute values in Tab. 4.1 for all the analysed configurations, *I-IX*. Using the notation in the comparison with previous work, these confusion matrices are obtained with $N_b = 1$

From comparison between these confusion matrices computed in [5] and the percentage values in Tab. 4.2 (considering only 61 MHz and 71 MHz) obtained with the CFD and ANN implemented for the proposed algorithm, these results either outperform or are comparable with the ones in the literature, unless one specific case, for several reasons as discussed below. First of all, the percentage values are better or comparable in most of the cases. In addition, for the case of only bandwidths in the confusion matrices of Tab. 4.3, high PM and FR classification accuracies cannot simultaneously be obtained with the same N_b value; more specifically, good accuracy for PM classification is reached with high N_b, while low N_b values are recommended for better FR classification accuracy. While, in case of both bandwidths and magnitudes, results that could be comparable with the percentage values in the proposed algorithm are reached with $N_b = 5$. This produces an increased complexity of their algorithm and computational time as shown in [5]. Moreover, it should be noticed that by increasing N_b the amount of testing bursts is dramatically reduced (from 200 to 40) which may reduce the reliability of the obtained classification rates. While, in

61-71 MHz - only bandwidths [5]

SIGNAL \ CLASS	FR	PM	FR	PM	FR	PM
FR	95.5%	4.5%	59%	41%	15%	85%
PM	35%	65%	7.6%	92.4%	0%	100%
no. of bursts (N_b):	*1*		*3*		*5*	

a)

61-71 MHz - both bandwidths and magnitudes [5]

SIGNAL \ CLASS	FR	PM	FR	PM	FR	PM
FR	100%	0%	100%	0%	100%	0%
PM	38.5%	61.5%	18.2%	81.8%	7.5%	92.5%
no. of bursts (N_b):	*1*		*3*		*5*	

b)

TABLE 4.3: Confusion matrices with percentage values corresponding to the results in [5] obtained with ED-based classifier. Two different feature variables are considered: a) only bandwidth of each signal and b) bandwidth + magnitude for each signal. Bursts corresponding to two transmission frequencies, 61-71 MHz, are used together as input data. Each of these two tables consists of three different levels (1, 3, and 5) for the amount of bursts averaged (N_b) to increase the frequency resolution

this work (which would correspond to $N_b = 1$), all the 1000 samples are firstly used to perform the testing step with better reliability of classification results and, secondly, high FR classification and PM classification are simultaneously obtained for each of the analysed configurations.

The only configuration in the results of the proposed work that produces minor PM classification accuracy than the other configurations, and then slightly smaller accuracy than the corresponding results for ED-based classifier, is $VIII$ which corresponds to -3 dBm at 61 MHz. This could be due to a possible reduced quality of the SBW waveform in the newly created dataset for this configuration; this could worsen the performance of the classifier.

Nevertheless, even if the SNR level becomes low, the performance relative to configurations form IV to IX is still sufficient to guarantee good accuracy of the classification rate. It is also worth noticing that, unlike the previous work in [5], in this work the proposed feature-based classifier is validated on a dynamic environment in which signals change either their carrier frequency or their transmission power (or both of them) in different cognitive cycles. Consequently, the training step is performed on a more complex dataset with a wider range of signal characteristics and shapes than in the previous work employing ED-based classifier in an almost

-3 dBm / 51 MHz [6]

SIGNAL \ CLASS	FR	PM	FR	PM	FR	PM	FR	PM
FR	99.1%	0.9%	99.8%	0.2%	98.5%	1.5%	99.1%	0.9%
PM	0%	100%	0%	100%	4.0%	96%	0%	100%
	i		*ii*		*iii*		*iv*	

TABLE 4.4: Confusion matrices in [6] for the parameter configuration -3 dBm / 51 MHz (corresponding to configuration *VII*, if notation in this work was considered). Combinations of the considered features are: i) bandwidth + amplitude + variance ii) bandwidth + amplitude iii) amplitude + variance iv) bandwidth + variance

stationary environment. This means that higher classification capability is required for dynamic scenarios.

In the literature, ref. [6] proposes an interference mitigation algorithm for Spectrum Intelligence employing a naive Bayes classifier to discriminate FR from PM waveforms in a stationary environment. In Tab. 4.4, the corresponding results are shown in percentage values. Specifically, the SBW signal, which is considered as PM waveform, is transmitted with onetenth (-3 dBm) power at 51 MHz central frequency.

Different combinations of available features are used to evaluate the performance of the classifier. Namely, four combinations of bandwidth, maximum magnitude, and variance for each of the extracted waveform (both PM and FR) from the spectrum of interest as shown in the confusion matrices in Tab. 4.4: i) classification rates when the classifier combines all the three features, ii), iii), and iv) classification rates when classifier combines bandwidth and maximum magnitude, maximum magnitude and variance, and bandwidth and variance, respectively.

The naive Bayes classifier is trained with 50 bursts while additional 50 bursts, independent from the training data, are used for the testing step. For each burst there is just one PM signal (the SBW) and several FR waveforms. Consequently, the classifier is tested on 50 PM samples and almost 2000 FR samples.

By comparison with the results in the proposed work with onetenth power and 51 MHz, in the FR classification case the CFD and ANN-based algorithm produces better or comparable percentage classification rates than the naive Bayes-classifier in Tab. 4.4 for all the four combinations of features. Concerning the PM classification, the performance of the algorithm for configuration *VII*, namely -3 dBm / 51 MHz, is not as good as in the other configurations and, consequently, lower values when compared with the percentage rates in Tab. 4.4. This has been discussed previously for configuration *VIII* and may be due to a possible reduced quality of the SBW waveform. In addition, it is worth noticing that for classifier in [6], only 50 samples

corresponding to SBW waveform in a stationary environment are used both during training step and during testing step which could produce unreliably classification rates. Again, dynamic scenarios require higher classification capability than stationary ones, which means that results in this work could be underestimated with respect to the previous work.

Considering that, the newly implemented *analysis* block in the proposed work shows promising performance for the CSI processing in complex dynamic environments that also include signals with low SNR level.

To conclude, a new AI-enabled cognitive radio framework is proposed as part of a *Cyclic Spectrum Intelligence algorithm* to enable interference mitigation in wideband radios. The CSI algorithm is based on a Cognitive Cycle consisting of 5 blocks: *Sensing*, *Processing*, *Analysis*, *Learning*, and *Acting*. The Cognitive Cycle interacts with the radio environment through the *sensing* block which obtains instantaneous spectrum data in the band of interest and the *acting* block whose task is to decide and change the transmission parameters such as the carrier frequency and the transmission power. The main novelty introduced by the proposed work is the *analysis* block. After having identified occupied channels and spectrum holes in the observed wideband signal, the stationary spectral analysis is performed, based on the cyclostationary feature of modulated signals, to extract the α-profile from the SCF of the detected signals. Afterwards, the extracted α-profile of a number of samples of the different waveforms detected in the wideband spectrum are fed to a pre-trained ANN to classify the waveforms as FR or PM. A software defined radio testbed has been employed to generate an experimental dateset and validate the *analysis* block. Results show high classification rate in most of the configurations even for low transmission power case; this makes the new block based on cyclostationary feature detector with artificial neural network a promising methodology for the CSI processing in complex dynamic environments that also include signals with low SNR level. In support of this claim, a comparison with existing works has also been conducted. The main drawback related to these two previous approaches is that the variability of both the hand-crafted features and noise does not guarantee accurate classification especially at low SNR level. On the contrary, the proposed approach overcomes that thanks to better extracted features and the neural network classifier. It has been shown that, the algorithm for dynamic environments proposed in this work produces better (in some cases) and comparable (in other cases) results than the ED- and naive Bayes-based algorithms. In addition, although the proposed CFD and ANN-based algorithm is trained and tested on a large number of samples from a dynamic scenario where higher classification capability is required, it still produces better or comparable classification rates with respect to the

existing approaches investigated in a stationary spectrum.

Future work includes analysis of fully autonomous systems capable of dynamically access the spectrum in a cognitive radio framework for applications such as PHY-layer security against jamming attacks. The effective applicability of the proposed approach to signals in a spectrum beyond 120 MHz and to different spectrum bands can also be investigated, although it is expected that the shape of the α-profile of a signal is independent of its fundamental frequency. In addition, Stealthy Jammer Detection Algorithm investigated in [20] and in [21] can be validated on the experimental dataset described in this work.

To further proceed toward the main objective of this thesis, *learning of dynamic models* should be considered. To this end, in the remainder of this chapter, different representations for dynamic signals are described in Sec. 4.3 through probabilistic graphical models used to represent single entity state and interacting entities situation assessment. Learning dynamic Bayesian representations is then presented in detail in Sec. 4.4 including techniques in the state-of-the-art employed to learn dynamic models for both single and interacting entities. To this end, several current techniques and algorithms, with some probable future work, are discussed in this context. Subsequently, a discussion about cognitive dynamic jamming can be found in Sec. 4.5. Finally, some future directions are introduced in in Sec. 4.6.

4.3 Probabilistic dynamic signal representation

Let us refer to the framework, presented in Sec. 3.5, in a dynamic scenario; the first step towards learning dynamic models is signal representation through feature-based, graphical, and probabilistic methods. This is a decisive step because the performance of the learning process is strongly influenced by the chosen signal representation. Consequently, features representing the signal of interest, the corresponding model, and the necessary parameters should be selected carefully since a sensible choice of them may produce better learning accuracy.

Features are extracted from the signal and represented in the corresponding feature space. Some of the extracted features can provide information about the state variable of the entity whose state needs to be dynamically estimated. They could be, e.g., central frequency, bandwidth, transmitting power, shape, and other more complex entity state descriptors (of analog and digital modulated signals), as presented in this section.

Consequently, a probabilistic representation of random variables related to the signal as random process can be obtained by using the extracted features. Specifically,

dynamic models (which includes the state variable and its noisy observations) are described by PGMs such as DBNs from Bayesian theory.

In the following sections, probabilistic models corresponding to both single and interacting entities are discussed along with some learning techniques for such kind of models.

4.3.1 Single entity state

To represent the state of each entity, *probabilistic graphical models (PGMs)* provide a graph-based representation as in [15]. The main idea is to encode complex distributions over high-dimensional spaces where causality links are defined among model components. In the basic graphical representation depicted in Fig. 4.4(a), random variables of the system model such as state and observables are represented by nodes, while edges (or links) express direct probabilistic relationships between them. Temporal dependencies among nodes can also be represented (often by horizontal edges) as can be seen later in this section. Furthermore, graphical models also introduce a compact representation for independencies among variables in the distribution described by corresponding graph. Considering that, probability distribution functions (PDFs) are often in the form of some parametric functions, the probabilistic relationship described by $p(\mathbf{x}|\mathbf{y}) = f(\mathbf{x}, \mathbf{y}, \theta)$ is represented as in Fig. 4.4(b) where $\mathbf{x}$ could be the state (hidden variable) of a dynamic system, $\mathbf{y}$ the corresponding observable, and θ the unknown parameter.

PGMs provide a statistical framework to model interactions and cause-effect relationships like in interaction analysis, and enable formalizing and handling the uncertainties. The basic idea is to provide a graphical tool to decompose a multivariate probability distribution into a factored form by providing an intuitive and manageable visual description.

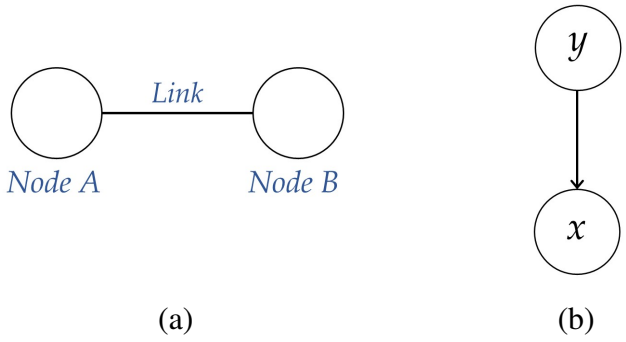


FIGURE 4.4: (a) Basic Probabilistic Grafical Model structure consisting of two nodes connected through a link; (b) graphical representation of the conditional probability $f_{\mathbf{X}|\mathbf{Y}}(\mathbf{x}|\mathbf{y})$

- *Dynamic bayesian networks (DBNs)* are based on Bayesian Network (BN) approach which produces a graph model describing the statistical relationships among a group of n random variables $\mathbf{X} = \{X_i\}_{i=1,2,\ldots,n}$. A BN is determined by its graph structure G and distribution parameter Θ. A variable X_i is independent of its non-descendants given all its parents $Pa(X_i)$ in G. Therefore, the joint probability distribution over $\mathbf{X}$ can be decomposed by:

$$Pr(\mathbf{X}) = \prod_{i=1}^{n} Pr(X_i | Pa(X_i)). \tag{4.1}$$

The parameter set $\Theta = \{\theta_i\}_{i=1,2,\cdots,n}$ specifies the parameters of each conditional distribution in Eq. (4.1). A Dynamic Bayesian Network (DBN), is the extension of a BN to model temporal processes. In DBN, a set of random processes are represented by the $\mathbf{X}(k) = \{X_i(k)\}_{i=1,2,\cdots,n}$, and $X_i(k)$ is the random variable of process at discrete time k. The network structure G now defines the dependency among variables over a period of time as well as those within the same time epoch. Assuming Markovian and causal processes, a node in graph G is only linked from the other nodes in the same or previous epoch [16]. Thus

$$Pr(\mathbf{X}(k+1) | \mathbf{X}(0:k)) = Pr(\mathbf{X}(k+1) | \mathbf{X}(k)), \quad k = 0, 1, 2, \ldots \tag{4.2}$$

Some basic dynamic models which can be represented by DBNs are now introduced.

- A *Hidden Markov Model (HMM)* is a tool for representing probability distributions over sequences of observations and is, in fact, a special case of the more general DBNs [13]. In Markov models, the state is directly visible to the observer, and therefore, the state transition probabilities are the only parameters to be considered whereas, in a HMM, the state is not directly visible, but only outputs, dependent on the state, are visible. The HMM assumes that the observation at time k was generated

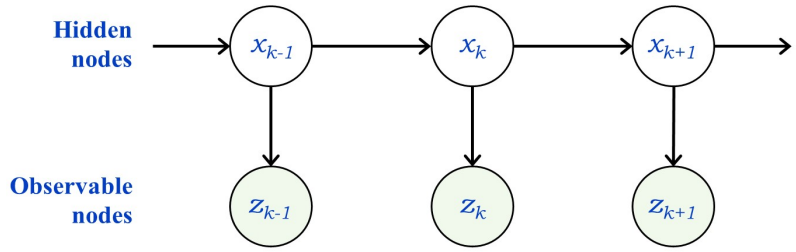


FIGURE 4.5: Hidden Markov Model representation: **x**'s are latent variables, **z**'s are the observable variables, while the horizontal arrows describe temporal dependencies

by some process whose state $\mathbf{x}_k$ is hidden from the observer. It also assumes that the state of this hidden process satisfies the Markov property, which is, given the value of $\mathbf{x}_{k-1}$, the current state $\mathbf{x}_k$ is independent of all the states prior to $k-1$. Graphically, it can be explained as shown in Fig. 4.5. The graph shows the dependencies between the variable of the model. $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \cdots, \mathbf{x}_k\}$ is a sequence of unobservable states and $\mathbf{Z} = \{\mathbf{z}_1, \mathbf{z}_2, \mathbf{z}_3, \cdots, \mathbf{z}_k\}$ is a sequence of observable emissions. Considering that the probability of being in a particular state at step i is known once the state at step $i-1$ is known and that the probability of seeing a particular emission at step i is known once the state step i is known, the joint distribution of a sequence of states and observations can be factored in the following way:

$$P(\mathbf{x}_{1:k}, \mathbf{z}_{1:k}) = P(\mathbf{x}_1) P(\mathbf{z}_1|\mathbf{x}_1) \prod_{i=2}^{k} P(\mathbf{x}_i|\mathbf{x}_{i-1}) P(\mathbf{z}_i|\mathbf{x}_i) \tag{4.3}$$

- A *Switching Linear Dynamical System (SLDS)* is defined in [11] by:

$$\begin{cases} \mathbf{x}_k = \mathbf{A}^{(a_k)}\mathbf{x}_{k-1} + \mathbf{v}_k^{(a_k)} \\ \mathbf{z}_k = \mathbf{C}\mathbf{x}_k + \mathbf{w}_k \end{cases} \tag{4.4}$$

with $a_k|a_{k-1} \sim \pi_{a_{k-1}}$. a_k belongs to a set $S \triangleq \{1, 2, \cdots, s\}$ consisting of a finite number of modes (i.e., categorical variable). When a discrete first-order Markov chain a_k with transition probabilities $\{\pi_{ij}\}$, $i, j \in S$, indexes the mode-specific linear dynamic system at time index k driven by Gaussian noise $\mathbf{v}_k^{(a_k)} \sim \mathcal{N}\left(0, \Sigma^{(a_k)}\right)$, it is called Jump Markov Linear System (JMLS). A JMLS can be seen as an extension of the HMM, which has the same mode evolution, but conditionally independent observations. The corresponding probabilistic graphical representation is shown in Fig. 4.6. *Rao-Blackwellized particle filter* is an example of a commonly used filter associated with a switching model.

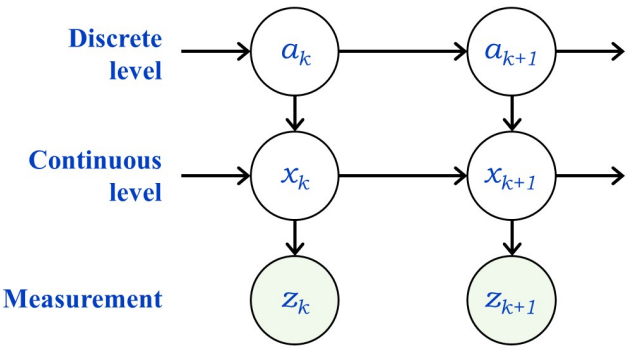


FIGURE 4.6: A SLDS representation in which **a**'s are the categorical variables

In general, the filtering process is performed to filter (estimate) and predict variables belonging to the dynamic models including nodes and links. Some common filters are described below.

- The *Kalman filter (KF)* provides optimal finite-dimensional algorithm for recursive Bayesian state estimation in linear-Gaussian cases. A linear dynamic system is described by the following state-space equations:

$$\mathbf{x}_k = \mathbf{F}_{k-1}\mathbf{x}_{k-1} + \mathbf{n}_{k-1} \tag{4.5}$$

$$\mathbf{z}_k = \mathbf{H}_k\mathbf{x}_k + \mathbf{w}_k \tag{4.6}$$

Eq. (4.5) represents the *process model* while Eq. (4.6) represents the *measurement model*. The Kalman filter assumes that the posterior pdf $p(\mathbf{x}_k|\mathbf{z}_k)$ at every step is Gaussian and then it can be completely characterized by the mean and the covariance. These assumptions hold if $\mathbf{n}_{k-1}$ and $\mathbf{w}_k$ are drawn from Gaussian density and the dynamic system is linear.

The matrices $\mathbf{F}_k$ and $\mathbf{H}_k$ define the linear functions in the dynamic system. Random sequences $\mathbf{n}_k$ and $\mathbf{w}_k$ are mutually independent zero-mean white Gaussian with covariance $\mathbf{Q}_k$ and $\mathbf{R}_k$, respectively. Mean and covariance predictions are given by:

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{F}_{k-1}\hat{\mathbf{x}}_{k-1|k-1} \tag{4.7}$$

$$\mathbf{P}_{k|k-1} = \mathbf{Q}_{k-1} + \mathbf{F}_{k-1}\mathbf{P}_{k-1|k-1}\mathbf{F}_{k-1}^T \tag{4.8}$$

respectively. In the update process the prediction is compared to the observation which results in the estimated mean and covariance are as follows:

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k\left(\mathbf{z}_k - \mathbf{H}_k\hat{\mathbf{x}}_{k|k-1}\right) \tag{4.9}$$

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{K}_k\mathbf{S}_k\mathbf{K}_k^T = \left[\mathbf{I} - \mathbf{K}_k\mathbf{H}_k\right]\mathbf{P}_{k|k-1} \tag{4.10}$$

where $\mathbf{z}_k - \mathbf{H}_k\hat{\mathbf{x}}_{k|k-1}$ is called innovation and denoted as $\mathbf{v}_k$, $\mathbf{S}_k$ is its covariance, and $\mathbf{K}_k$ is the Kalman gain. In the presence of a high Kalman gain, i.e. when the $\mathbf{P}_{k|k-1}$ is large or the $\mathbf{S}_k$ is small, the innovation is considered as reliable. A large Kalman gain value occurs when the prediction is not consistent and/or the innovation is trustworthy and it implies that the estimate $\mathbf{x}_{k|k}$ relies more on the innovation than on prediction.

- The *Extended Kalman Filter (EKF)* is a suboptimal solution of the Bayesian filtering problem for non-linear systems with additive noise

$$\mathbf{x}_k = \mathbf{f}_{k-1}\left(\mathbf{x}_{k-1}\right) + \mathbf{n}_{k-1} \tag{4.11}$$

$$\mathbf{z}_k = \mathbf{h}_k\left(\mathbf{x}_k\right) + \mathbf{w}_k \tag{4.12}$$

The basic idea is to linearise the non-linear functions by the first term in the Taylor series expansion. Prediction and innovation are computed using, respectively, non-linear functions $\mathbf{f}_{k-1}$ and $\mathbf{h}_k$.

By substituting $\mathbf{f}_{k-1}$ and $\mathbf{h}_k$ with their local linearisation (first order Taylor approximation):

$$\hat{\mathbf{F}}_k = \left[\nabla_{\mathbf{x}_{k-1}} \mathbf{f}_k^T\left(\mathbf{x}_{k-1}\right)\right]^T \Big|_{\mathbf{x}_{k-1}=\hat{\mathbf{x}}_{k-1|k-1}} \tag{4.13}$$

$$\hat{\mathbf{H}}_k = \left[\nabla_{\mathbf{x}_k} \mathbf{h}_k^T\left(\mathbf{x}_k\right)\right]^T \Big|_{\mathbf{x}_k=\hat{\mathbf{x}}_{k|k-1}} \tag{4.14}$$

where

$$\nabla_{\mathbf{x}_k} = \left[\frac{d}{d\mathbf{x}_k\left[1\right]} \quad \cdots \quad \frac{d}{d\mathbf{x}_k\left[n_x\right]}\right]^T \tag{4.15}$$

and $\mathbf{x}_k\left[i\right]$ is the i-th component of the state vector, a linearised Kalman Filter is obtained as in Eqs.(4.5) - (4.10).

The Unscented Kalman Filter (UKF), the Cubature Kalman Filter (CKF), and the Gauss-Hermite Kalman Filter (GHKF) are also variations of KF which handle the non-linear Gaussian problem. Furthermore, non-linear filtering also include *particle filter and its variants*.

4.3.2 Interacting entities situation assessment

The previous section investigated single entity representation through PGMs which well capture dependencies of variables. More specifically, *temporal dependencies* (dynamics of the system) and *entity-to-entity dependencies* (interactions among objects) can be represented in graphical models.

To introduce entity-to-entity dependencies, interaction oriented DBN structures can be employed with linked nodes belonging to different entities (interacting entities). Specifically, the DBN structure corresponding to a system consisting of two interacting objects described through a SLDS model (introduced in the previous section) is shown in Fig. 4.7.

Such a kind of links define the probability that a random variable of an entity influences (or is influenced by) one or more random variables of a different entity.

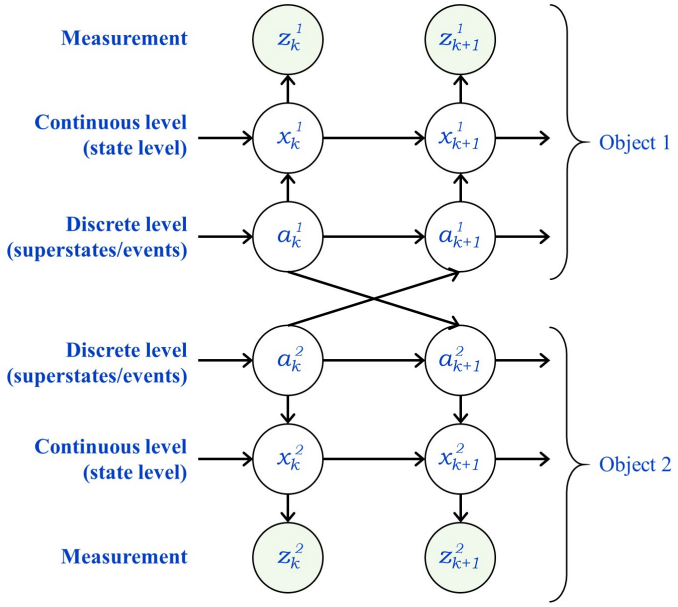


FIGURE 4.7: Interaction oriented SLDS structure with linked nodes belonging to two interacting objects

Interactions may also happen among three or more objects each of them can be represented through PGMs.

From neuroscience, cognition and interaction are two inter-related functionalities of biological systems. A *bio-inspired* approach is the main idea towards the concept of cognitive dynamic systems (CDSs) and the development of interactive systems [9]. Dynamic systems take into account embodied and situated cognition by adaptively changing their state. The goal of dynamic systems is to maintain stability of the equilibrium between the object and the environment (i.e. maintenance of the proper level of security and/or safety). CDSs build up rules of behaviour over time through learning from continuous experiential interactions with the environment, and thereby deal with environmental uncertainties.

Specifically, a CDS can be described as a system whose design closely mimics the human brain and is motivated by human cognition. Cognitive reasoning is based on dynamical dispositional representations of the interactions between an object in contextual scenario and the changed organism state it causes [7]. Such a dispositional representation of external objects with respect to a self-object is the key aspect of the knowledge embedded on bio-inspired CDSs. The capability of *learning from experience* and the idea of *autobiographical memory (AM)* are drawn from the bio-inspired approach of the interactions occurring between the system and the user. Indeed, this concept is based on neurophysiological observations of human brain structure for

modelling and learning interactions between the user and the system and provides engineering implications in the development of context-aware learning and predicting strategies.

Self-awareness plays a key role in the development of *self-adaptation* techniques. In particular, self-adaptation based on self-awareness at the individual level means that one single entity receives inputs both from itself or some of its components and from the external environment and uses the input to adjust to the current conditions [17]. *Multiple entity adaptation* is then obtained by introducing the interaction of different self-adaptation techniques at the level of the single individual.

4.4 Learning dynamic Bayesian representations

Learning *dynamic models* can be considered a main objective in CR applications where the spectrum of interest hosts several signals whose parameters (such as central frequency, transmit power, modulation scheme, and so forth) may change across the measurement time. To this end, *learning from available data* (current and past data) should in perspective become a major approach used in this framework. Basically, in order to learn a model which describes a *single entity*, parameters are estimated and predicted by using statistical signal processing techniques and Bayesian filters. The cycle observation-update-prediction performs learning of parameters and dynamic models at successive time instants. In other words, when the system collects a new observation of the state, both the predicted transition probability of the state and the predicted measurements (obtained at the previous step) are updated with the new data and their predictions for the subsequent time instant are also computed, as introduced in the KF description, for example. Again, when a new observation is collected, a new cycle observation-update-prediction is performed.

In this section, after introducing basic concepts of learning, several techniques employed to learn causal conditioned probabilities, dynamic models, and interactions are described along with some current work and probable future directions in the field of learning dynamic models.

4.4.1 Learning vocabulary, state and state changes

A learning approach in a Bayesian framework starts with some a priori knowledge about both probabilistic distributions of the model structure (namely edges in the graphical model) and the model parameters. A-prior probability distributions over model structures and model parameters represent this initial knowledge which is then updated using the data to obtain the corresponding posterior probability distribution

over models and parameters [12]. Denoting the prior distribution over model structures with $P(\mathbf{M})$, the priori distribution over parameters for each model structure with $P(\theta|\mathbf{M})$, and the data-set as $\mathbf{Z} = \{\mathbf{z}_1, \mathbf{z}_2, \cdots, \mathbf{z}_t\}$, the posterior distribution over models can be obtained through Bayes rule as follows:

$$P(\mathbf{M}|\mathbf{Z}) = \frac{\int P(\mathbf{Z}|\theta, \mathbf{M})\, P(\theta|\mathbf{M})\, d\theta P(\mathbf{M})}{P(\mathbf{Z})} \tag{4.16}$$

which takes into account the uncertainty in the parameters. The posterior distribution over the parameters, for a given model structure, can be obtained through:

$$P(\theta|\mathbf{M}, \mathbf{Z}) = \frac{P(\mathbf{Z}|\theta, \mathbf{M})\, P(\theta|\mathbf{M})}{P(\mathbf{Z}|\mathbf{M})} \tag{4.17}$$

Now, based on available data and models, the next observation $\mathbf{z}_{t+1}$ can be computed by Bayesian prediction as:

$$P(\mathbf{z}_{t+1}|\mathbf{Z}) = \int P(\mathbf{z}_{t+1}|\theta, \mathbf{M}, \mathbf{Z})\, P(\theta|\mathbf{M}, \mathbf{Z})\, P(\mathbf{M}|\mathbf{Z})\, d\theta d\mathbf{M} \tag{4.18}$$

based on averages over both the uncertainty in the model structure and in the parameters. This is known as *predictive distribution* for each model.

Typically, the state space model (SSM) of a *dynamic system* consists of three random processes; namely, the *state model*, the *process model* of the state of a signal (or entity) which describes the transition of the state, and the *measurement model* which describes observations of the state:

$$\begin{aligned} \mathbf{x}_0 &\sim p_0(\mathbf{x}_0) \\ \mathbf{x}_t|\mathbf{x}_{t-1} &\sim p_f(\mathbf{x}_t|\mathbf{x}_{t-1}) \\ \mathbf{z}_t|\mathbf{x}_t &\sim p_g(\mathbf{z}_t|\mathbf{x}_t) \end{aligned} \tag{4.19}$$

where $\mathbf{x}_t$ and $\mathbf{z}_t$ are the state and measurement vectors at time t, p_0 is the initial state probability distribution function (PDF), p_f is a conditional probability density function (CPDF) representing the dynamics of the state and p_g is a CPDF representing the measurement process. Interaction-oriented models include linked dynamic systems, as described in Sec. 4.3.2, to represent *interacting entities* in an ICE scenario.

Changes in the state of a dynamic system, described by the corresponding transition probability, are referred to as *trajectories*. Each entity adopts a strategy inside the spectrum of interest, which defines its behaviour. *Normal behaviour* means that it

is predictable because already observed, and consequently the implemented model is capable of obtaining accurate estimates (and predictions) of the parameters and model by using learned information gathered from past experience. When the hypothesized model does not produce the estimates (and predictions) accurately, the behaviour is said to be *abnormal*. This happens when the strategy has not been observed in past experiences, or from a security point of view, that behaviour is different (or not allowed) from legitimate behaviour.

4.4.2 Learning causal conditioned distributions

Learning techniques can be classified in *supervised learning* based on labelled data but it is always an expensive training algorithm and difficult to obtain, *unsupervised learning* which does not need the labelled data making it a better training algorithm. On the other hand, *semi-supervised learning* is a class of ML techniques which is receiving increasing interest in the last decade. These techniques combine both labelled and unlabelled data items in their training process. Therefore, they are usually applied in data sets in which only a small subset of data items may be effectively labelled, due to the high costs and time required in the labelling process [1]. In this section, some methods used in the literature to learn model parameters are introduced.

- *HMMs* are one of the most important techniques to model and classify sequential data with several applications in sequence modelling problems like speech recognition, Human Activity Recognition (HAR), or time series analysis [22]. The Expectation-Maximization (EM) algorithm is the classical method used to learn the parameters of HMMs. However, it exhibits two main problems: 1) the likelihood is multimodal so the EM is guaranteed to converge only to a local maxima, and 2) the multiple initializations required for minimizing the effects of the local convergence and the more than quadratic growth with the number of hidden states makes EM computationally heavy with large training dataset. Bayesian inference methods including Gibbs sampling, variational optimization, or Bayesian non-parametric methods are even computationally heavier and global convergence is still not guaranteed. A spectral algorithm for learning HMMs with discrete observations in proposed in [14]. This method adjusts the model by moment matching instead of maximizing the likelihood, and it relies on the use of the observable operators view of the HMM [22].

Learning of SLDSs is introduced in the practical example of Sec. 4.5. Now, two supervised learning methods are briefly described. While some unsupervised techniques are discussed is Sec. 4.4.4.

- *Support Vector Machines (SVMs)* are supervised learning models with associated learning algorithms that analyse data used for classification and regression analysis. The SVM method for regression is formulated in solving a convex optimization problem, more specifically a quadratic programming (QP) problem [8].

In SVM for non-linear function regression, the main idea is to approximate the dataset

$$D = \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_k, y_k), \cdots, (\mathbf{x}_N, y_N)\}, \qquad \mathbf{x}_k \in R^n, \qquad y_k \in R \tag{4.20}$$

with a non-linear function

$$f(\mathbf{x}) = < \boldsymbol{\omega}, \phi(\mathbf{x}) > + b \tag{4.21}$$

where $< \cdot, \cdot >$ denotes the dot product; $\boldsymbol{\omega} \in R^{n_h}$ is the weight vector in primal weight space; $\phi(\cdot) : R^n \rightarrow R^{n_h}$ is the non-linear function that maps the input space to a high dimensional feature space where linear regression is performed; b is the bias term. The optimization problem is given by:

$$\min \frac{1}{2}\|\boldsymbol{\omega}\|^2 + C \sum_{k=1}^{N} \left(\xi_k + \xi_k^*\right) \qquad s.t. \qquad \begin{cases} y_k - < \boldsymbol{\omega}, \phi(\mathbf{x}) > - b \leq \varepsilon + \xi_k \\ < \boldsymbol{\omega}, \phi(\mathbf{x}) > + b - y_k \leq \varepsilon + \xi_k^* \\ \xi_k, \xi_k^* \geq 0 \end{cases} \tag{4.22}$$

where ε is the approximation accuracy that can be violated by means of the slack variables ξ, ξ^* for the non-feasible case. The constant $C > 0$ determines a trade-off between the flatness of f and the amount up to which deviations larger than ε are tolerated. A smaller value of C tolerated a larger deviation. From the constrained optimization problem in Eq. 4.22, the Lagrange function can be written as in [26] with multipliers α and α^*. Consequently, the resulting SVM takes the form:

$$f(\mathbf{x}) = \sum_{k-1}^{N} \left(\alpha_k - \alpha_k^*\right) < \phi(\mathbf{x}), \phi(\mathbf{x})_k > + b \tag{4.23}$$

where the inner product $< \phi(\mathbf{x}), \phi(\mathbf{x})_k >$ can be defined through a kernel $K(\mathbf{x}, \mathbf{x}_k)$.

However, the major drawback of SVM is its higher computational burden because of the required constrained optimization programming. Major breakthrough has been obtained at this point with a least squares version of SVM, called LS-SVM.

- *Gaussian Processes (GPs) for regression* is a powerful supervised learning algorithm well-suited to high dimensional data analysis and non-linear estimation problems [2]. Basically, given the covariance function, provides an analytical solution to any regression estimation problem. It does not only provide point estimates, but

it also gives confidence intervals for them. In GPs for regression, the optimization step to set the hyperparameters of the covariance function is performed by maximum likelihood. Any GP method assumes a zero-mean GP prior over the space of possible functions and a Gaussian likelihood model. The posterior can be analytically computed, it is a Gaussian density function as well as the predictions given by the model.

Given a labelled training data-set ($D = \{\mathbf{x}_i, y_i\}_{i=l}^{n}$, where the input $\mathbf{x}_i \in \mathcal{R}^{d\times 1}$ and the output $y_i \in \mathcal{R}$) and a new input location $\mathbf{x}^*$, the probability distribution for its output y^*, i.e. $p(y^*|\mathbf{x}^*, D)$ can be predicted. Assuming a Gaussian linear prediction model for y_i : $p(y_i|\mathbf{x}_i, \mathbf{w}) = \mathcal{N}\left(y_i; \mathbf{w}^T\phi(\mathbf{x}_i), \sigma_v^2\right)$, where $\phi(\cdot)$ defines a transformation of the input space, and a zero-mean Gaussian prior over $\mathbf{w}$, $p(\mathbf{w}) = \mathcal{N}\left(\mathbf{w}; \mathbf{0}, \sigma_{\mathbf{w}}^2\mathbf{I}\right)$, the posterior for the weight vector w using Bayes theorem in given by:

$$p(\mathbf{w}|D) = \frac{p(y_i|\mathbf{X}, \mathbf{w})\, p(\mathbf{w})}{p(y_i|\mathbf{X})} = \mathcal{N}(\mathbf{w}; \mu_{\mathbf{w}}, \Sigma_{\mathbf{w}}) \tag{4.24}$$

where $\mu_{\mathbf{w}} = \Phi^T\Phi/\sigma_v^2 + \mathbf{I}/\sigma_{\mathbf{w}}^2$, $\mathbf{y} = [y_1, \cdots, y_n]^T$, $\Phi = [\phi(\mathbf{x}_1), \cdots, \phi(\mathbf{x}_n)]^T$, and $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_n]^T$. The prediction for y^* is obtained integrating out the posterior over $\mathbf{w}$ times its likelihood:

$$p(y^*|\mathbf{x}^*, D) = \int p(y^*|\mathbf{x}^*, \mathbf{w})\, p(\mathbf{w}, D)\, d\mathbf{w} = \mathcal{N}\left(y; \mu_{y^*}, \sigma_{y^*}\right) \tag{4.25}$$

where

$$\mu_{y^*} = \phi^T(\mathbf{x}^*)\, \mu_{\mathbf{w}} = \mathbf{k}^T\mathbf{C}^{-1}\mathbf{y} \tag{4.26}$$

$$\sigma_{y^*}^2 = \phi^T(\mathbf{x}^*)\, \Sigma_{\mathbf{w}}\phi(\mathbf{x}^*) = k(\mathbf{x}^*, \mathbf{x}^*) + \mathbf{k}^T\mathbf{C}^{-1}\mathbf{k} \tag{4.27}$$

being $k\left(\mathbf{x}_i, \mathbf{x}_j\right) = \phi^T(\mathbf{x}_i)\, \phi^T\left(\mathbf{x}_j\right)$, $(\mathbf{C})_{ij} = k\left(\mathbf{x}_i, \mathbf{x}_j\right) + \frac{\sigma_\mu^2}{\sigma_w^2}\delta_{ij}$, and $\mathbf{k} = [k(\mathbf{x}^*, \mathbf{x}_1), \cdots, k(\mathbf{x}^*, \mathbf{x}_n)]$. To obtain the estimation given by a GP model for regression, it is necessary only to specify its covariance function $k(\cdot, \cdot)$.

4.4.3 Learning interactions

A decision support-based system should be capable of inferring on occurring behaviours and interactions of each entity in the corresponding environment. Behaviours are the specific activities defined as actions of each individual object without any external influence. While interactions are the actions inducted by pairwise exchanges of influences between the objects [3].

Basically, coupled DBNs are introduced as appropriate models to represent interactions between the entities [4]. Indeed, probabilistic processes and graphical models are characterised by their capability of working in the presence of uncertainty and noise in the environment of interest as well as with a large number of inter-related variables. In addition to these intrinsic capabilities of DBNs, the computational load is kept to a manageable level even with co-existence of many entities and models.

A probabilistic model based on a specific type of event takes inspiration from a bio-inspired approach and the concept of AM as described in Sec. 4.3.2. Damasio describes the cognitive entities as complex systems with incremental learning capabilities based on experience of the interactions between themselves and the external world [7]. Two specific brain processes can be defined to formalize the above concept called proto-self and core-self.

The DBN model proposed in [10] takes into account conditioned probability densities (CPDs) to represent these concepts. They are relative to both the state of each individual object in the environment (regardless of the presence of other objects) and to the interactions between two objects defined in terms of casual events. In particular, two conditioned probabilities describe the probability that the event in the core-self node produces the event in the proto-self node and the probability that the event in the proto-self node provokes the event in the core-self node. Causal relationships between the two entities are then described by CPDs which consider the interactions between the two objects. In other words, the entity's initial state, an external stimulus (the cause) and its consequence on the behaviour of the entity (the effect) can learn an entity's most frequent reaction to the action of another element in the scene.

Several widely used dynamic probabilistic methods in the literature can be found in [24].

4.4.4 Some current learning techniques and probable future directions

Deep learning has dramatically improved the state-of-the-art in many different ML topics like object detection, speech recognition, and machine translation. Following the initial development of neural networks, the recent success of deep learning is due to its deep architecture based on the idea of a system that simulates human brain [27]; this allows deep learning to solve many more complicated tasks.

The main objective is now to maintain the strength of neural networks while reducing the number of computation units (or neurons). Indeed, it has been shown that the representation of a $(k-1)$-layer neural network with exponentially many neurons is equivalent to a k-layer structure with polynomial many neurons.

In the framework of deep generative models, two examples that have been proposed recently are *generative adversarial networks* (GANs) and *variational autoencoders* (VAEs). Typically, learning the underlying data distribution of unlabelled signals or images can be highly challenging and inference on such distributions is highly computationally expensive or intractable. GANs and VAEs provide efficient approximations, making it possible to learn tractable generative models of unlabelled data [25]. Recent works have also extended the VAE and the GAN from unsupervised to semi-supervised settings.

A theoretical framework about GANs and VAE is provided in Sec. 5.6. Other learning techniques could also be considered from the *Game theory* framework such as *Pursuit evasion game* and *Multi-armed bandit*.

4.5 A discussion of application directions of ML techniques to cognitive dynamic jamming

A possible shared radio environment, as basis for future ICE, is depicted in Fig. 4.8 in which there are several interacting agents such as a jammer, which aims to "catch" legitimate signals, and a user whose objective is to avoid the jammer. In addition, a network manager (NM) is an external observer of the spectrum, e.g., a primary base station, that monitors dynamical spectrum joint situations generated by actions of both the jammer and the user. Such a station can interact with the user and could be either a human operator or an ICE system. The NM can be useful for example in case the spectrum sensing computational resources are too intensive to be implemented directly in the user device, and the spectrum monitoring facility is shared among many users of the network. Furthermore, in case the NM is the only CDS, it can

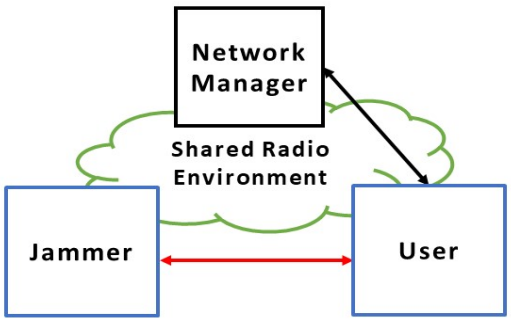


FIGURE 4.8: A practical example in a shared radio environment consisiting of two interacting agents, a jammer and a user, with opposite objectives and a network manager. The NM interacts with the user

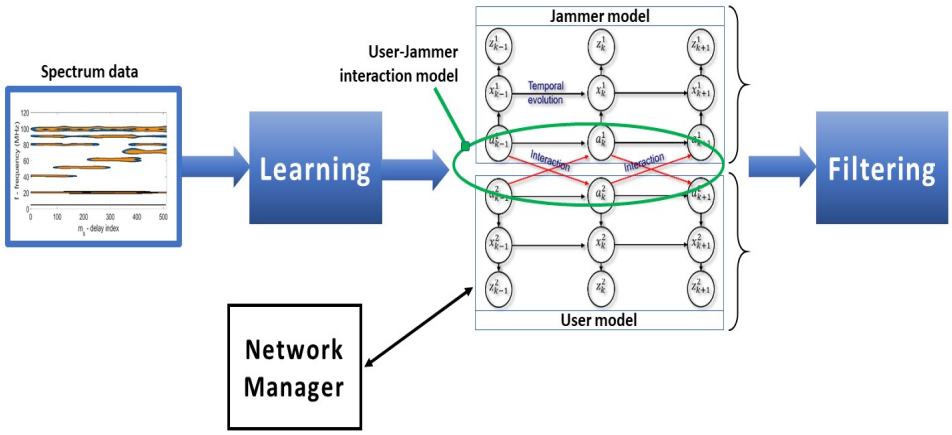


FIGURE 4.9: The proposed example in which jammer, user, and their interactions are modelled through JMSs. The NM observes the spectrum and interacts with the user. Learning and filtering blocks are also shown in the picture

be in charge to continuously check the normality of an observed situation, namely to predict if the user behaviour is in line with avoiding the jammer.

In general, each of the three radios representing the user, the jammer or the NM, can be a cognitive radio that can be modelled as a CDS. In this example, the NM is assumed to be a CDS that use two JMSs to model the observed interaction between user and jammer as in Fig. 4.9. The goal is to provide a discussion on how this model can be learned from observed spectrum behaviours where the user successfully avoids the jammer. Such data series correspond to normality patterns that can be used as examples from which to train NM's CDS. First of all, features coming from spectrum monitoring methods are extracted, e.g. by using the time-frequency representation of the spectrum of interest as described in Sec. 3.5. Such spectrum data can be used by the *learning* algorithm to estimate parameters of the switching models. In other words, normality models are learned corresponding to models where rules that allow the user to avoid the jammer are estimated from the data series examples where this is known to have happened successfully. Such rules describe a dynamic equilibrium condition that should be verified according to previous experiences recorded in such data series if a normal situation occurs. Normal training samples are much easier to collect than abnormality samples whose dataset could be insufficient or unavailable [23]. After having trained the switching models using only normal data in order to learn a representation of the normal spectrum activities, the *filtering* phase produces predictions of the corresponding hidden variables and observation at each time instant. Filtering can be also thought as a generative block for abnormality detection; indeed, since the JMSs are not able to generate abnormal events at testing time

because trained with only normal data; a normality condition is said to be probabilistically verified if updates and related predictions are consistent. On the contrary, an excess of deviations of updated observations from predictions correspond to abnormalities where the jammer's behaviour is different from the patterns observed during the training and the user is no longer capable of avoiding it.

From Eq. (4.4), the state vector $\mathbf{x}_k^i$ for the i-th entity can be augmented into $\tilde{\mathbf{x}}_k^{i\,T} = \left[\mathbf{x}_k^{i\,T}, \mathbf{a}_k^{i\,T}\right]$, where $i = \{1, 2\}$, then the corresponding non-linear state-space model (NLSSM) is given by:

$$\begin{cases} \tilde{\mathbf{x}}_k^i = f\left(\tilde{\mathbf{x}}_{k-1}^i, \mathbf{v}_k^i\right) \\ \mathbf{z}_k^i = h\left(\tilde{\mathbf{x}}_k^i, \mathbf{w}_k^i\right) \end{cases} \tag{4.28}$$

The NLSSM for interactive entities evolves according to

$$\Pi\left(\mathbf{a}_k^i | \mathbf{a}_{k-1}^i\right), \quad p\left(\mathbf{x}_k^i | \mathbf{x}_{k-1}^i, \mathbf{a}_k^i, \mathbf{a}_{k-1}^j\right), \quad p\left(\mathbf{z}_k^i | \mathbf{x}_k^i, \mathbf{a}_k^i, \mathbf{a}_{k-1}^j\right); \quad i, j = \{1, 2\}, \; i \neq j \tag{4.29}$$

and [10]

$$\Pi\left(\mathbf{a}_k^1 | \mathbf{a}_{k-1}^2\right), \quad \Pi\left(\mathbf{a}_k^2 | \mathbf{a}_{k-1}^1\right) \tag{4.30}$$

Specifically, in this example *learning* aims to estimate (from available data) model parameters, continuous- and discrete-valued states, transition and interaction probability distributions in non-linear state space models with Markovian switching structure. Discrete switching variables ($\mathbf{a}$'s), probabilistic distributions described by links between the discrete nodes (namely the $\mathbf{\Pi}$ matrices) including entity-to-entity interactions, and probabilistic distributions described by links between discrete variables and continuous state variables ($\mathbf{x}$'s) can be learnt through unsupervised clustering techniques such as *Dirichlet Process Mixture (DPM)* model. On the other hand, continuous state variables, transition models between the state at time k and its value at time $k-1$, and likelihood models between observations ($\mathbf{z}$'s) and state variables can be learnt through techniques like *Gaussian Process (GP) regression* or *GANs*.

The *filtering* block estimates and predicts sequentially the latent states $\left\{\mathbf{x}_k^i, \mathbf{a}_k^i\right\}$ given the densities for the initial state $\left\{\mathbf{x}_0^i, \mathbf{a}_0^i\right\}$ and the measurements up to time k ($\mathbf{z}_{1:k}^i$). For this purpose, *particle filtering*-based methods can be used such as *Markov Jump Particle Filter* (MJPF) or *Rao-Blackwellized Particle Filter* (RBPF).

4.6 Conclusion and future directions

This chapter addresses the application of learning techniques to dynamic models for wireless communications. Indeed, PHY-layer security is gathering ever growing interest in the research world due to the high vulnerability of wireless communications to external attacks such as jamming attacks. This is the main objective for which a detailed discussion of selected techniques in the current state-of-the-art is provided throughout the sections above. Some probable and interesting future directions in this framework are also introduced in order to show the evolution of CR, which has been conceived to overcome the shortage of available bands in the wireless spectrum, towards a wider paradigm designed for interactive and cognitive environments. Considering that, *dynamic* and *interaction* are decisive concepts to enable cognition which is on the basis of self-aware (and self-adaptive) devices. To this end, time-information seems to be a key component in processing dynamic signals (in Chapter 3, time-frequency analysis was employed to detect signals in the spectrum of interest and to extract both time and frequency information). Several statistical models are described, most of them are based on a Bayesian approach and represented through probabilistic graphical models which highlight both temporal dependencies, namely dynamics of the system, and entity-to-entity dependencies, namely interactions.

A more practical point of view is described in the example about application directions of ML techniques for cognitive dynamic jamming which lays the foundations towards complex CDSs to make ICEs become a reality.

In particular, probable future directions include employing techniques and algorithms used for data analysis, image and video processing, robotics, and so forth, in wireless communications. Indeed, general concepts such as entity, state, and trajectory can be specified to CR as signal, central frequency and bandwidth, and time-frequency information, respectively. Specifically, representation of wireless signals (including jammers) through, for example, multilayer perceptron, variational autoencoders, or HMMs are not found in the current state-of-the-art and, consequently, learning techniques to learn such kind of dynamic models in a jamming context are not either.

In the following chapter, recent developments on *learning dynamic models* following the *data-driven approach with deep network architectures* are presented as enablers of *self-aware systems*. Indeed, the main objective of current research is to build AI-based CR devices. In particular, a bio-inspired approach, based on deep networks, could enhance the learning process by combining *cognition with self-awareness and self-adaptation* in CR devices for future PHY-layer security.

Bibliography

[1] F. A. Breve and D. C. G. Pedronette. Combined unsupervised and semi-supervised learning for data classification. In *2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6, September 2016.

[2] S. Caro, F. Perez-Cruz, and J. J. Murillo-Fuentes. Gaussian processes for regression in channel equalization. In *14th European Signal Processing Conference (EUSIPCO 2006), Florence, Italy*, September 2006.

[3] F. Castaldo, F. A. N. Palmieri, and C. S. Regazzoni. Bayesian analysis of behaviors and interactions for situation awareness in transportation systems. *IEEE Transactions on Intelligent Transportation Systems*, 17(2):313–322, February 2016.

[4] S. Chiappino, L. Marcenaro, P. Morerio, and C. Regazzoni. Run length encoded dynamic bayesian networks for probabilistic interaction modeling. In *21st European Signal Processing Conference (EUSIPCO 2013)*, pages 1–5, September 2013.

[5] K. Dabčević, M.O. Mughal, L. Marcenaro, and C. S. Regazzoni. Spectrum intelligence for interference mitigation for cognitive radio terminals. In *Wireless Innovation Forum European Conference on Communications Technologies and Software Defined Radio (WInnComm- Europe), Rome, Italy*, November 2014.

[6] K. Dabčević, M.O. Mughal, L. Marcenaro, and C. S. Regazzoni. Cognitive radio as the facilitator for advanced communications electronic warfare solutions. *Journal of Signal Processing Systems*, 83(1):29–44, April 2016.

[7] A. Damasio. *The Feeling of What Happens: Body and Emotion in the Making of Consciousness*. In: Harvest Books, October 2000.

[8] S. Dhar and V. Cherkassky. Universum learning for svm regression. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 3641–3648, May 2017.

[9] A. Dore, A. F. Cattoni, and C. S. Regazzoni. Interaction modeling and prediction in smart spaces: A bio-inspired approach based on autobiographical memory. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 40(6):1191–1205, Nov 2010.

[10] A. Dore and C. Regazzoni. Interaction analysis with a bayesian trajectory model. *IEEE Intelligent Systems*, 25(3):32–40, May 2010.

[11] E. Fox, E. B. Sudderth, M. I. Jordan, and A. S. Willsky. Bayesian nonparametric inference of switching dynamic linear models. *IEEE Transactions on Signal Processing*, 59(4):1569–1585, April 2011.

[12] Z. Ghahramani. An introduction to hidden markov models and bayesian networks. *International Journal of Pattern Recognition and Artificial Intelligence*, 15(01):9–42, 2001.

[13] S. R. Gomes, S. G. Saroar, M. Mosfaiul, A. Telot, B. N. Khan, A. Chakrabarty, and M. Mostakim. A comparative approach to email classification using naive bayes classifier and hidden markov model. In *2017 4th International Conference on Advances in Electrical Engineering (ICAEE)*, pages 482–487, September 2017.

[14] D. Hsu, M. K. Sham, and T. Zhang. A spectral algorithm for learning hidden markov models. *Journal of Computer and System Sciences*, 78(5):1460–1480, 2012. JCSS Special Issue: Cloud Computing 2011.

[15] D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques - Adaptive Computation and Machine Learning*. The MIT Press, 2009.

[16] Y. Liu and B. Cai. A reliability analysis framework based on time-varying dynamic bayesian network. In *2015 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, pages 21–25, December 2015.

[17] M. Maggio, T. Abdelzaher, L. Esterle, H. Giese, J.O. Kephart, O.J. Mengshoel, A.V. Papadopoulos, A. Robertsson, and K. Wolter. *Self-adaptation for Individual Self-aware Computing Systems*, pages 375–399. In: Self-Aware Computing Systems, Springer International Publishing, Cham, 2017.

[18] M.F. Møller. A scaled conjugate gradient algorithm for fast supervised learning. *Neural networks*, 6(4):525–533, 1993.

[19] T. Nawaz, D. Campo, M.O. Mughal, L. Marcenaro, and C.S. Regazzoni. Jammer detection algorithm for wide-band radios using spectral correlation and neural networks. In *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC), Valencia, Spain*, June 2017.

[20] T. Nawaz, L. Marcenaro, and C. S. Regazzoni. Defense against stealthy jamming attacks in wide-band radios: A physical layer approach. In *IEEE Global Conference on Signal and Information Processing (GlobalSIP' 17), Montreal, Canada*, October 2017.

[21] T. Nawaz, L. Marcenaro, and C.S. Regazzoni. Stealthy jammer detection algorithm for wide-band radios: A physical layer approach. In *IEEE 10th International Workshop on Selected Topics in Wireless and Mobile computing (STWiMob' 17), Rome, Italy*, October 2017.

[22] A. Nazábal and A. Artés-Rodríguez. Discriminative spectral learning of hidden markov models for human activity recognition. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1966–1970, April 2015.

[23] M. Ravanbakhsh, M. Nabi, E. Sangineto, L. Marcenaro, C. S. Regazzoni, and N. Sebe. Abnormal event detection in videos using generative adversarial nets. *2017 IEEE International Conference on Image Processing (ICIP)*, pages 1577–1581, 2017.

[24] H. I. Suk, B. K. Sin, and S. W. Lee. Analyzing human interactions with a network of dynamic probabilistic models. In *2009 Workshop on Applications of Computer Vision (WACV)*, pages 1–6, December 2009.

[25] C. Wan, T. Probst, L. V. Gool, and A. Yao. Crossing nets: Combining gans and vaes with a shared latent space for hand pose estimation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1196–1205, July 2017.

[26] H. Wang and D. Hu. Comparison of svm and ls-svm for regression. In *2005 International Conference on Neural Networks and Brain*, volume 1, pages 279–283, October 2005.

[27] H. Wang and B. Raj. On the origin of deep learning. *ArXiv e-prints*, February 2017.

5 AI principles and Deep Generative Models

As mentioned in Sec. 2.1, CR has recently been defined to exhibit three integral attributes which are observations, reconfiguration, and cognition. Learning and reasoning are the fundamental aspects of cognition but cognition cannot be achieved until and unless CR network subsumes intelligence, which can be developed by implementing AI techniques [7] following a data-driven approach. In particular, *data-driven self-awareness in CR based on deep learning is the major innovation provided in the remaining part of this thesis*. For this reason, an overview of related concepts from neuroscience along with the motivation of deep learning can be found in the following sections. Spectrum abnormality detection in CR is then briefly described. Finally, three recent deep generative models (C-GAN, AC-GAN, and VAE) are presented as part of the proposed research in the learning block (**stage 2**) of the diagram of Fig. 5.1.

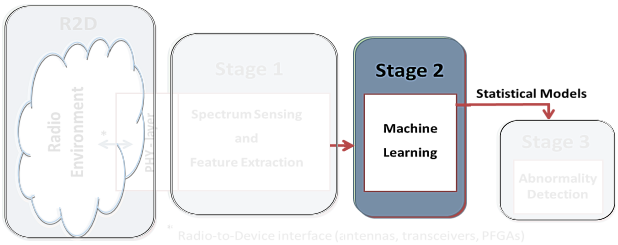


FIGURE 5.1: Learning phase (stage 2)

5.1 AI basis: self-aware systems and the free-energy principle

Self-awareness [32]: in the scientific community, the concept of fundamental characteristic of autonomic systems is defined as "*to be autonomic, a computing system needs to know itself.*" In particular, the autonomic computing literature, which made large efforts to provide systems with greater levels of autonomy and the ability to manage themselves [27], proposes four supporting properties of autonomic systems [9]: self-awareness, environment-awareness (or self-situation), self-monitoring and self-adjustment. All of them have been a focus not only of artificial intelligence researchers, but computer engineers have also identified these basic concepts as part of future systems that use run-time knowledge about themselves [32].

To this end, the notion of computational self-awareness has been introduced as interpretation of self-awareness in computing systems inspired by human self-awareness in psychology. In [37], self-awareness assumes a twofold meaning. *Explicit SA* defined as "*the capacity to become the object of one's own attention*" and based on the ability to consider oneself as an object or entity within the world, and *implicit SA* which is subjective where the individual is aware of its experiences within the world. These experiences are private to the individual, and are typically not externally observable.

Considering that, self-aware computing systems are computing systems that [29]:

1. learn models capturing knowledge about themselves and their environment (such as their structure, design, state, possible actions, and runtime behaviour) on an ongoing basis and
2. reason using the models (e.g., predict, analyze, consider, and plan) enabling them to act based on their knowledge and reasoning (e.g., explore, explain, report, suggest, self-adapt, or impact their environment) in accordance with higher-level goals, which may also be subject to change.

Recently, it has been suggested that important processes such as self-awareness can be explained within a free-energy framework [13]. As described in the following paragraph, this could be a result of the brain's attempts to minimise the amount of free-energy (or 'surprise') in sensory systems in order to be in states where the environment is highly predictable through the optimisation of predictions about the sensory consequences of events occurring in the environment [2].

The free-energy principle [2]: to have an idea about the principle, let us mention the sentence in [19] "*biological agents resist a natural tendency towards disorder in a constantly changing environment*". Basically, the brain (as the organ within an agent that evaluates information about the external and internal states and resists disorder) must have a low level of entropy (entropy being the surprise averaged over all events encountered) [19]. To do this, the brain only needs to minimise surprise associated with the current event by making predictions about what sensorial consequences will be evoked by events in the environment. Predictions are updated and optimised continuously over time in order that a low level of entropy is maintained across the brain. In the long-term, this means that the brain as a whole minimises the average of surprise in all sensory systems, learning how best to model and predict incoming sensory input. Additionally, it means that short-term phasic surprises ('prediction errors'), which are processed locally at each node of each sensory system, are avoided by actions that minimise surprise. Free-energy acts as the upper bound on the level of surprise, which necessitates that surprise is minimised in two ways [20, 18].

Firstly, agents can act upon the environment to alter the incoming sensory events in a manner that minimises prediction errors namely minimises surprise across the brain as actions with surprising sensory outcomes are avoided and prediction errors are kept at a low level. Secondly, prediction errors can cause agents to update estimates about the causes of the sensory events [6, 19], in order to obtain optimal inferences about the actual causes of sensory events. Expectations are made prior to any event based on representations of the probability of a sensory event occurring and, when there is a sensory event which is discrepant from the expected input, the prediction errors cause an update of the prior expectations dynamically. In this way, with the updated estimates of the priors of the causes of a sensory event, future expectations are modified in a way that similar sensory events in the future are predicted [14, 16].

The brain is therefore dynamically processing shifting generative models about what is causing incoming sensory events, based on probabilistic predictions about how likely something is to have happened and what the likely causes are.

The interested readers are invited to read Damasio [8], Friston [40, 17, 20, 15], Haykin [26, 25, 24], and Penrose [23] for more information about the immense universe around understanding biological brain and artificial intelligence in a multidisciplinary basis. In particular, Friston introduces the *concept of free energy minimization* based on *generalised states* in neuroscience.

This is the reason a *generalized state vector* is investigated in this thesis as representation of the instantaneous spectrum. In this way, abnormality indicator, Receiver Operating Characteristic curve, and Area Under ROC Curve/Accuracy are metrics used to obtain a certain measure of self-awareness of the radio spectrum, as shown in Chapter 7.

To conclude, the innovation lies in a fact that *self-awareness can be implemented through generative models (even in CR)* especially thanks to the recent advances in multi-layers networks architectures in the context of deep learning, as shown in the following sections.

5.2 AI and Self-Awareness in CR

Cognitive radio (CR) was introduced in 1999 by ref. [36] for a flexible spectrum access where CR is defined as the integration of model-based reasoning with software radio technologies [35]. In 2005, a review of the CR concept is given and treated as brain-empowered wireless communications [24]. CR is a radio or system that senses the environment, analyzes its transmission parameters, and then makes decisions to improve the utilization of the radio electromagnetic spectrum and maintain highly

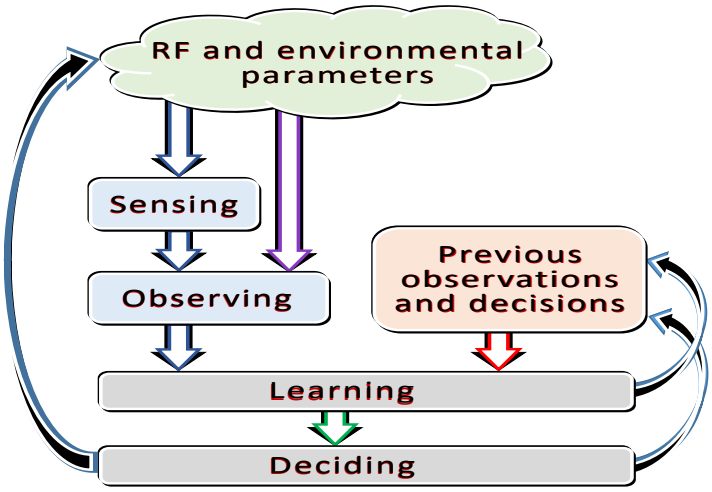


FIGURE 5.2: Learning loop in cognitive radios [1]

reliable communications. In this way, CR can be an intelligent wireless communication system that is aware of its environment by using methodologies to learn from the environment and adapt to statistical variations in the input stimuli [24].

Consequently, CR devices need to be equipped with learning and reasoning abilities where the cognitive engine needs to coordinate the actions of the CR by making use of AI and deep learning techniques. In this context, self-awareness has been recognised as a valuable property of CR [32].

The learning loop in cognitive radios is shown in Fig. 5.2 which can provide a certain degree of SA and can be presented as follows: (1) sensing the radio frequency (RF) parameters, (2) observing the environment and analyzing its feedback, (3) learning, (4) keeping the decisions and observations for updating the model and obtaining better accuracy in future decision-making, and finally (5) deciding on abnormality situations [5, 42].

In [46], the CR is introduced from the perspectives of AI which may be represented but not limited to the following learning techniques: fuzzy logic, genetic algorithms, neural networks, game theory, reinforcement learning, support vector machine, case-based reasoning, decision tree, entropy, Bayesian, Markov model, multi-agent systems, and artificial bee colony algorithm. The various cognitive radio network tasks need different learning techniques, such as supervised/unsupervised learning and single-agent/multi-agent, to learn and adapt to any change in the environment. In addition, a CR may not have any prior knowledge of the operating RF environment such as noise, interference, and spectrum activity. Therefore, the most suitable learning technique depends on (1) the available information, (2) spectrum characteristics, and (3) the CR task and problem to address.

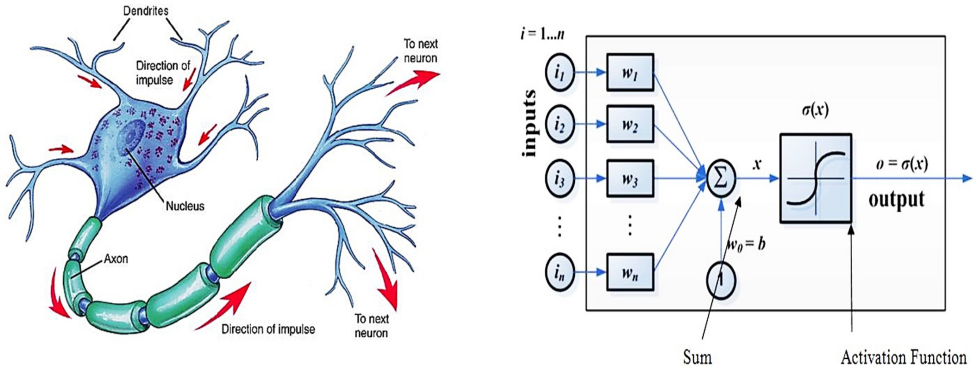
In any case, it is possible to take advantage from deep learning techniques which

is the objective of this work. Deep architectures lay the basis for AI-based implementation of cognitive capabilities like self-awareness in CR devices. This work, where detection of abnormal activities inside the observed spectrum by means of deep generative models enhances the self-awareness capability, can be considered as a little step toward AI-enabled radio and network. A brief introduction to the concept of abnormality detection as an SA functionality in CR is now introduced.

5.3 Deep learning for data-driven SA

Recently, effective implementation of (data-driven) SA employs deep learning which is a set of methods that allows a machine to automatically discover the hidden representations of raw data for detection or classification purposes [30]. Conventional machine learning techniques were limited in their ability to process natural data in their raw form. Indeed, many existing machine learning algorithms use what are called shallow architectures, including neural networks with only one hidden layer, kernel regression, and support vector machines, among many others [43]. Theoretical results show that the internal representations learned by such systems are necessarily simple and are incapable of extracting certain types of complex structure from rich sensory input [4, 3]. Training these systems also requires large amounts of labeled training data.

By contrast, recent discoveries in neuroscience strongly suggest that building brain inspired structures or systems requires deep architectures, models composed of several layers of nonlinear processing. It appears that, for example, object recognition in the visual cortex uses many layers of non-linear processing and requires very little labeled input [31]. A basis representation of the biological neuron is shown in Fig. 5.3(a) along with the corresponding mathematical representation (perceptor) in



(a) (b)

FIGURE 5.3: Schematic representation of: a) a biological neuron [38] and b) the corresponding artificial neuron (perceptor) [11]

Fig. 5.3(b). Thus, development of new and efficient learning algorithms for models with deep architectures that can also make efficient use of a large supply of unlabeled sensory input is of crucial importance. An example of deep architecture is shown in Fig. 5.4. It is the ConvNet structure consisting on four convolutional layers as well as max-pooling layers.

Deep learning methods are representation learning methods with multiple levels of abstraction, obtained by composing simple but non-linear modules that each transform the representation at one level (starting with the raw input) into a representation at a higher, slightly more abstract level. With the composition of enough such transformations, very complex functions can be learned. In this way, deep learning allows computational models composed of multiple processing layers to learn representations of data with multiple levels of abstraction. Deep Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) are two examples of deep architectures that resulted in the rapid adoption of deep learning.

Deep learning is thus making major advances in solving problems following a data-driven approach that have resisted the best attempts of the AI community for many years. It has turned out to be very good at discovering intricate structures in high-dimensional data and is therefore applicable to many domains of science, business, and government. Indeed, *deep learning can effectively enable AI and data-driven SA*.

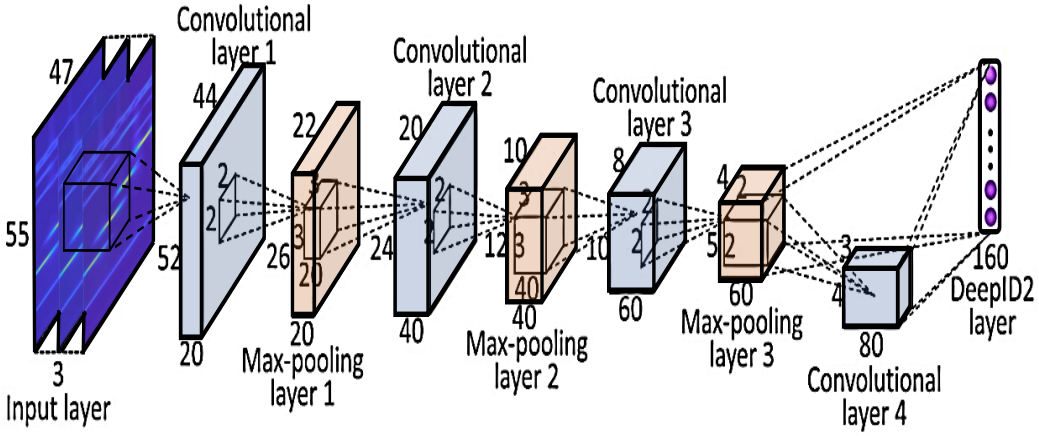


FIGURE 5.4: An example of deep network: the ConvNet structure for DeepID2 feature extraction [44]

The spectrum abnormality detection is now described in detail as part of the proposed basic SA module, along with the deep generative models employed to perform this task in CR devices. The abnormality measurement (or indicator) is defined for each of the three models.

5.4 Proposed SA functionality: spectrum abnormality detection

In the radio environment, normal spectrum activity or behaviour means that the signal is predictable because already observed and, consequently, the implemented model is capable of obtaining accurate estimates (and predictions) of the signal by using learned information gathered from past experience. Such a model describes a dynamic equilibrium condition that should be verified according to previous experiences recorded in such data series if a normal situation occurs.

Consequently, a CR device, which is transmitting in the radio spectrum and observing simultaneously the environment, can notice by using the learnt model any deviation from the normal situation (i.e. similar to what it was learned from previous experience) because the hypothesized model does not produce the estimates (and predictions) accurately. This happens, for example, when the strategy has not been observed in past experiences or from a security point of view, that behaviour is different (or not allowed) from legitimate behaviour.

The detected differences (called abnormalities) can be used either by the control system to apply abnormality mitigation strategies or by the SA module itself to incrementally learn new models that describe different dynamic equilibrium situations not included in previous experiences. *This contributes to an innovative Cognitive PHY-layer security in wireless communications*.

From the system point of view in Fig. 5.5, normality models are learned during the *training phase* by estimating the model parameters from the observed normality spectrum where rules or activities follow normality patterns. After having trained

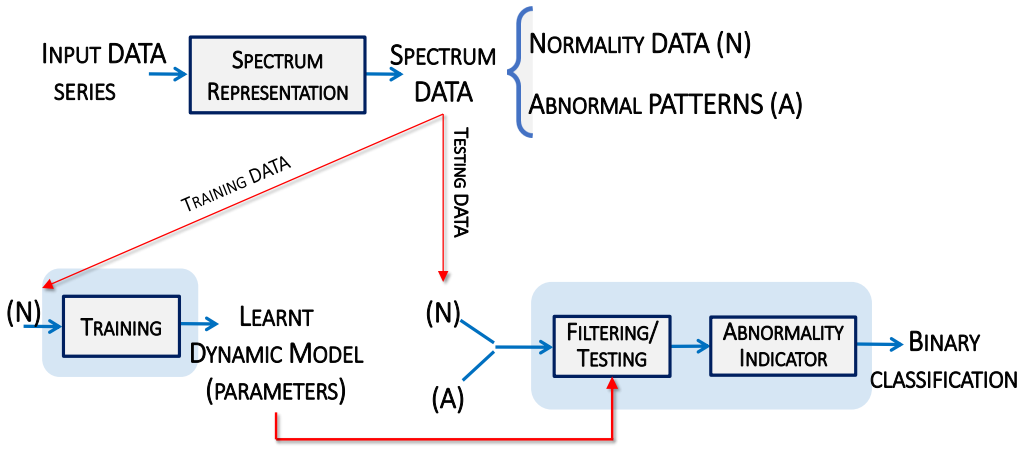


FIGURE 5.5: General scheme for learning

the models using only normal data, predictions of variables and observations are produced from the learned representation at each time instant during the *testing phase*. A *generative model* can be used for abnormality detection; since the learnt model is not able to generate approximations of abnormal events at testing time (because trained with normal data only), a normality condition is said to be probabilistically verified if updates and related predictions are consistent, while excess of deviations of updated observations from predictions corresponds to abnormalities where the signal's behaviour is different from the patterns observed during the training.

In the following sections, a detailed description of the generative models used in the proposed research is introduced. While, the implemented abnormality detection through generative models is presented in Chapter 6.

5.5 Deep generative models for spectrum abnormality detection

In the last few years, many of the most interesting advancements in the field of learning have come through novel applications of deep learning to generative modeling tasks. By definition, *a generative model describes how a dataset is generated, in terms of a probabilistic model. By sampling from this model, it is possible to generate new data* [12]. Indeed, generative models encode full probability distributions and specify how to generate data that fit such distributions. In other words, such models can infer relevant structure from the data which consists of many features. The learnt model can generate new sets of features that look as if they have been created using the same rules as the original data. The model must include a stochastic (random) element that influences the individual samples generated by the model.

Mathematically, in the generative modeling framework [12]:

- let us assume that the observations $\mathbf{x}$ have been generated according to some unknown distribution $p_{\boldsymbol{\theta}}(\mathbf{x})$, where $\boldsymbol{\theta}$ is the parameter vector of the data distribution;
- a generative model $q_{\boldsymbol{\phi}}(\mathbf{x})$ with parameter vector $\boldsymbol{\phi}$ tries to approximate $p_{\boldsymbol{\theta}}(\mathbf{x})$. If successful, $q_{\boldsymbol{\phi}}(\mathbf{x})$ can be sampled to generate observations that appear to have been drawn from $p_{\boldsymbol{\theta}}(\mathbf{x})$.

Generative modeling is usually performed with an unlabeled dataset (that is, as a form of unsupervised learning), though it can also be applied to a labeled dataset to learn how to generate observations from each distinct class. In this case, a generative model that estimates the distribution $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{y})$ should be built.

The objective is reached when both the model can generate examples that appear to have been drawn from $p_{\boldsymbol{\theta}}(\mathbf{x})$ and the model can generate examples that are suitably different from the observations $\mathbf{x}$. In other words, the model should not simply reproduce things it have already seen.

Generative modeling challenges include how the model copes with the high degree of conditional dependence between features and how the model finds one of the tiny proportion of satisfying possible generated observations among a high-dimensional sample space. Deep learning is the key to solving both of these challenges.

Some *deep architectures* employed in the proposed research are now discussed along with some details on how they are applied as abnormality detector.

5.6 GAN, C-GAN, AC-GAN, and VAE

Deep learning aims at representing probability distributions through hierarchical models, which has been witnessed to produce promising results over data encountered in AI applications [21, 3]. Recent research deploys deep convolutional networks (CNNs) to form new architectures where any pooling layers are replaced with strided convolutions and fractional-strided convolutions (for example, in networks representing a generator, a discriminator, an encoder, and a decoder) while fully connected hidden layers are removed for deeper architectures. In this way, convolutional structures achieve high sample quality and high training stability. In this thesis, three generative models are investigated and, then, compared.

Basic GANs: as shown in Fig. 5.6, the generative adversarial networks (GANs) consist of both a generative model G, with model distribution p_g that captures the data distribution $p_{data}(\mathbf{x})$, and a discriminative model D that learns to determine whether a sample is from p_g or $p_{data}(\mathbf{x})$ by estimating the probability $D(\mathbf{x})$ that a sample comes from the data distribution rather than the model distribution [21]. The training data is denoted as $\mathbf{x}$, the mapping to data space is represented by $G\left(\mathbf{z};\boldsymbol{\theta}_g\right)$ with parameters $\boldsymbol{\theta}_g$, $\mathbf{z}$ is random noise with prior $p_{\mathbf{z}}(\mathbf{z})$, while $D(\mathbf{x};\boldsymbol{\theta}_d)$ represents the discriminator with parameters $\boldsymbol{\theta}_d$ that outputs a single scalar. Both G and D can be represented by a non-linear mapping function such as a multilayer perceptron. The two models are simultaneously trained. The training procedure for G is to minimize the probability $log(1 - D(G(\mathbf{x})))$ that D makes correct decision. While D is trained to maximize the probability $log(D(\mathbf{x}))$ of correctly differentiating the training samples from generated samples. This framework corresponds to a two-player min-max game with cost function $V(G, D)$ given in [21]. In the space of arbitrary functions G and D, a unique solution exists, with G recovering the training data distribution and

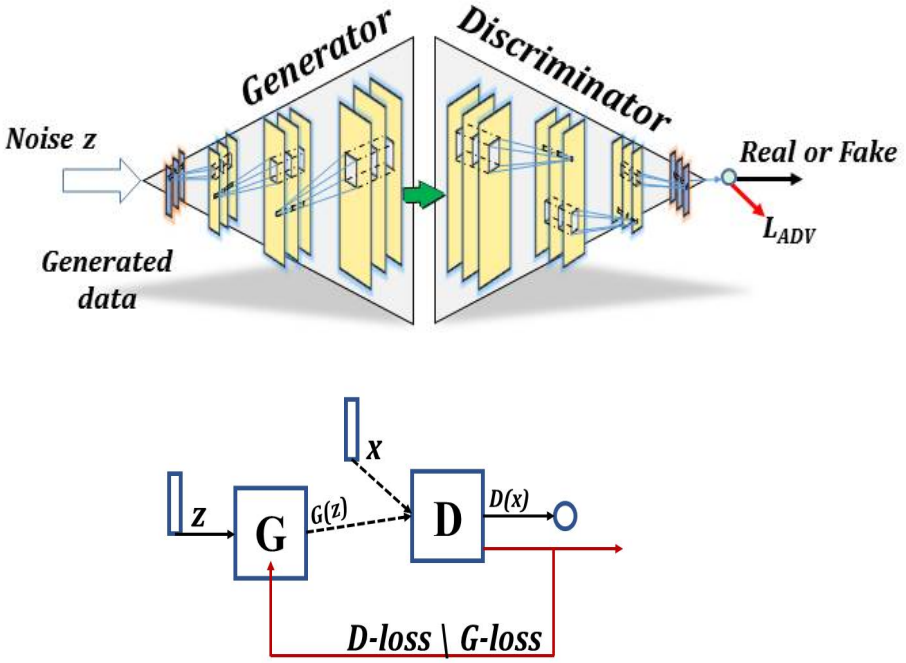


FIGURE 5.6: Deep GAN diagram

D equal to $1/2$ everywhere. In GANs, the objective function of the two-player game is given by:

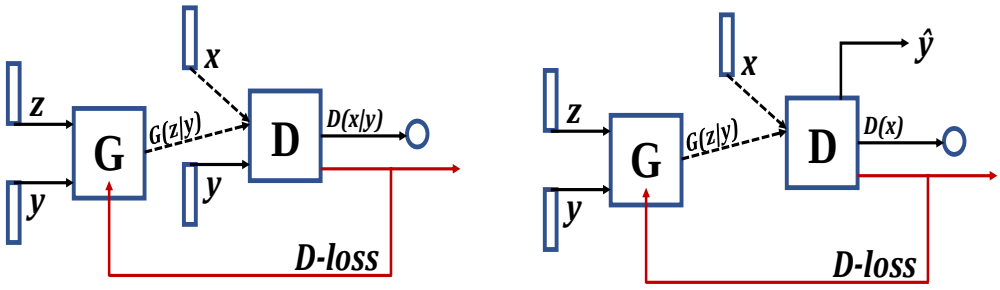
$$\min_G \max_D V(D,G) = \mathrm{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})}\left[\log D(\mathbf{x})\right] + \mathrm{E}_{\mathbf{z}\sim p_z(\mathbf{z})}\left[\log\left(1 - D(G(\mathbf{z}))\right)\right] \quad (5.1)$$

In an unconditioned generative model, there is no control on modes of the data being generated. However, by conditioning the model on additional information it is possible to direct the data generation process.

In the following, the two variants of the basic GAN model, C-GAN and AC-GAN [33], investigated in the proposed framework are described in details.

Conditional GAN (C-GAN): By conditioning the basic GAN model on additional information $\mathbf{y}$ (e.g. class labels), it is possible to direct the data generation process. This model is called C-GAN [34] shown in Fig. 5.7(a).

Conditional GAN (C-GAN) - Training Phase: the C-GAN consists of both a generative model G that captures the data distribution and a discriminative model D that estimates the probability of a sample comes from that data distribution. Both G and D can be represented by a non-linear mapping function that is learnt during the training phase. G maps a random noise $\mathbf{z}$ to data space $\mathbf{x}$. This mapping is represented by $G(\mathbf{z}|\mathbf{y})$. While D acts as a binary classifier and outputs a single scalar represented by $D(\mathbf{x}|\mathbf{y})$. The training procedure for G is to minimize the probability that D makes a



(a) Diagram of the C-GAN

(b) Diagram of the AC-GAN

FIGURE 5.7: Generative Adversarial Networks

correct decision. While D is trained to maximize the probability of correctly differentiating the training samples from the generated samples. This framework corresponds to a two-player min-max game. The corresponding cost function is given by:

$$\min_G \max_D V(D,G) = \mathrm{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})}\left[\log D(\mathbf{x}|\mathbf{y})\right] + \\ + \mathrm{E}_{\mathbf{z}\sim p_z(\mathbf{z})}\left[\log\left(1 - D(G(\mathbf{z}|\mathbf{y}))\right)\right] \tag{5.2}$$

where $p_{data}(\mathbf{x})$ is the data distribution and $p_z(\mathbf{z})$ is the prior.
Due to the conditioning variable $\mathbf{y}$, C-GANs belong to the supervised learning family. Nevertheless, for the purpose of this work, no actual labels are assigned to $\mathbf{y}$ since a null value is constantly given regardless the input data (normality or abnormal). Considering that, the C-GAN is employed as *unsupervised learning* method because no priory information is required for each kind of input data.
Conditional GAN (C-GAN) - Testing Phase: the parameters of both G and D networks are not updated through the optimization of the cost function. Deviations between prediction and observation are detected based on the following anomaly measurement:

$$db0 = |l^{MSE}_{real} - l^{MSE}_{fake}| \tag{5.3}$$

where the general formula to represent the MSE loss for two vectors $\mathbf{u}$ and $\mathbf{v}$ is given by:

$$l^{MSE}(\mathbf{u},\mathbf{v}) = \frac{1}{N}\sum_{n=1}^{N}(u_n - v_n)^2 \tag{5.4}$$

Let $\mathbf{u}$ be the output of the discriminator, $\mathbf{v}$ the corresponding adversarial ground truth, and N the batch size (in this case, l^{MSE} is the adversarial loss). Specifically, $l^{MSE}_{real} = \frac{1}{N}\sum_{n=1}^{N}(\mathbf{u}_{n,\,real} - \mathbf{v}_{n,\,real})^2$ is the MSE loss computed at the discriminator when the input is the real data $\mathbf{x}$, while $l^{MSE}_{fake} = \frac{1}{N}\sum_{n=1}^{N}\left(\mathbf{u}_{n,\,fake} - \mathbf{v}_{n,\,fake}\right)^2$ is the MSE loss when the input is the one generated by the generator from G($\mathbf{z}|\mathbf{y}$). The

notation $|\cdot|$ represents the absolute value function. By sequentially computing the percentile value over a number of time instants (samples), where the difference in Eq. (5.3) is computed, an abnormality indicator ($db0$) can be obtained.

Auxiliary Classifier GAN (AC-GAN): Alternatively, the discriminator can be modified with reconstructing the class information $\hat{\mathbf{y}}$. In this way, the discriminator will contain an auxiliary decoder network that outputs the class label for the training data. This variant of the GAN architecture is called auxiliary classifier GAN (or AC-GAN) [39] and shown in Fig. 5.7(b).

Auxiliary Classifier GAN (AC-GAN) - Training Phase: G uses both the class labels $\mathbf{y}$ and the noise $\mathbf{z}$ to generate data samples (fake data), $\mathbf{x}_{fake}$. In this case, the discriminator computes both the probability distribution of the sources, $p(\mathbf{s}|\mathbf{x})$, and of the class labels, $p(\mathbf{y}|\mathbf{x})$ such that $D(\mathbf{x}) = (p(\mathbf{s}|\mathbf{x}), p(\mathbf{y}|\mathbf{x}))$. The source of the data, $\mathbf{s}$, refers to the decision of the discriminator, namely either real data, $\mathbf{s}_{real}$, or fake data, $\mathbf{s}_{fake}$. Consequently, the objective function consists of both the log-likelihood of the correct source, $L_{\mathbf{s}}$, and the log-likelihood of the correct class, $L_{\mathbf{y}}$, as follows:

$$L_{\mathbf{s}} = \mathrm{E}\left[\log p(\mathbf{s}_{real}|\mathbf{x}_{real})\right] + \mathrm{E}\left[\log p\left(\mathbf{s}_{fake}|\mathbf{x}_{fake}\right)\right] \tag{5.5}$$

$$L_{\mathbf{y}} = \mathrm{E}\left[\log p(\hat{\mathbf{y}}|\mathbf{x}_{real})\right] + \mathrm{E}\left[\log p\left(\hat{\mathbf{y}}|\mathbf{x}_{fake}\right)\right] \tag{5.6}$$

D maximizes the probability of correctly classifying real and fake samples ($L_{\mathbf{s}}$) and correctly predicting the class label ($L_{\mathbf{y}}$) of a real or fake sample ($L_{\mathbf{s}} + L_{\mathbf{y}}$). G minimizes the ability of the discriminator to discriminate real and fake samples while also maximizing the ability of the discriminator in predicting the class label of real and fake samples ($L_{\mathbf{y}} - L_{\mathbf{s}}$).

Auxiliary Classifier GAN (AC-GAN) - Testing Phase: as in C-GAN, in this phase the parameters of both G and D networks are not updated, and the anomaly measurement defined in Eq. (5.7) is utilized to detect deviations as follows:

$$db0 = |l_{real} - l_{fake}| \tag{5.7}$$

The general formula to represent the loss for two vectors $\mathbf{u}$ and $\mathbf{v}$ is given by:

$$l(\mathbf{u}, \mathbf{v}) = l^{L8} + l^{CE} = \frac{1}{N}\sum_{n=1}^{N}(u_n - v_n)^8 + l^{CE} \tag{5.8}$$

where l^{L8} is the $L8$-loss (adversarial loss) and l^{CE} is the Cross Entropy loss (auxiliary loss). As in C-GAN, l_{real} and l_{fake} can be computed from Eq. 5.8. By sequentially computing the percentile value over a number of time instants (samples), where the difference in Eq. (5.7) is computed, an abnormality indicator ($db0$) can be obtained.

Variational Auto Encoders (VAEs) - Training Phase: VAEs learn a stochastic mapping between an observed data space $\mathbf{x}$, whose empirical distribution is typically complicated, and a latent space $\mathbf{z}$, whose distribution can be relatively simple [28]. $\mathbf{z}$ represents a compressed low dimensional representation of the input $\mathbf{x}$. VAEs consist of two models, the encoder or inference model, and the decoder or generative model (refer to Fig. 5.8). The generative model (decoder) learns the joint distribution $p_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z})$. The inference model (encoder) $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$, approximates the true but intractable posterior $p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x})$ of the generative model. The model parameters of the decoder and encoder are denoted by $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$, respectively. While, $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ are the mean and standard deviation of the multivariate distribution $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$. $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is a noise random variable. Ideally, the reconstructed input $\mathbf{x}'$ is approximately identical to $\mathbf{x}$, $\mathbf{x} \approx \mathbf{x}'$. The distribution $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$ can be parameterized using deep neural networks. In this case, the variational parameters $\boldsymbol{\phi}$ include the weights and biases of the neural network.

VAEs provide a computationally efficient way for optimizing the generative model jointly with the corresponding inference model. The model parameters ($\boldsymbol{\phi}$), also called variational parameters, are optimized such that:

$$q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}) \approx p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x}) \tag{5.9}$$

by using the Evidence Lower Bound (ELBO) which is the variational lower bound

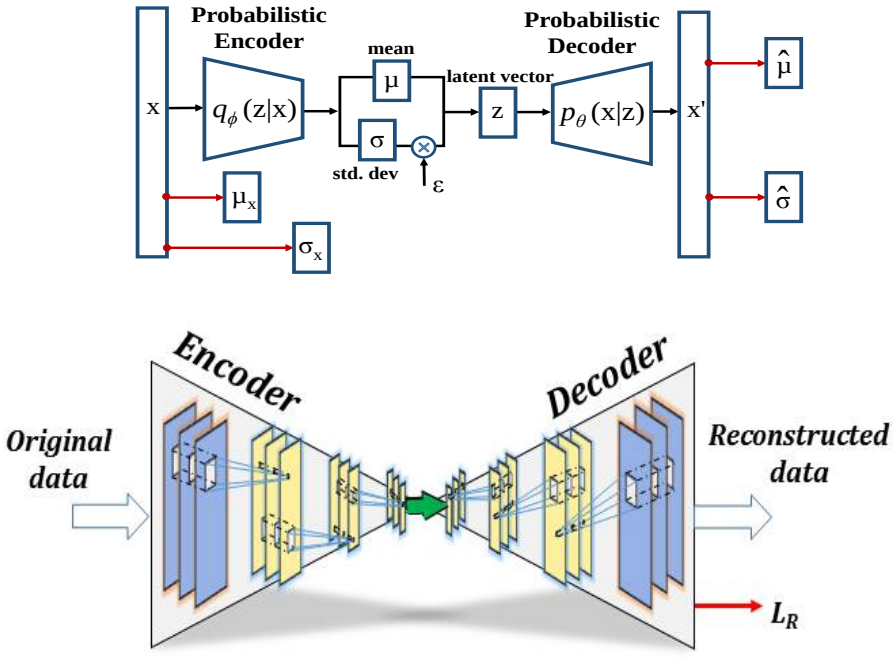


FIGURE 5.8: Diagram of the VAE

on the log-likelihood of the data given by:

$$\mathcal{L}_{\theta,\phi}(\mathbf{x}) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[\log p_\theta(\mathbf{x},\mathbf{z}) - \log q_\phi(\mathbf{z}|\mathbf{x})\right] = \log p_\theta(\mathbf{x}) - D_{KL}\left(q_\phi(\mathbf{z}|\mathbf{x})\,||p_\theta(\mathbf{x},\mathbf{z})\right) \tag{5.10}$$

The second term in Eq. (5.10) is the Kullback-Leibler (KL) divergence between $q_\phi(\mathbf{z}|\mathbf{x})$ and $p_\theta(\mathbf{x},\mathbf{z})$, which is non-negative. Due to the non-negativity of the KL divergence, the ELBO is a lower bound on the log-likelihood of the data:

$$\mathcal{L}_{\theta,\phi}(\mathbf{x}) \leq \log p_\theta(\mathbf{x}) \tag{5.11}$$

This lower bound $\mathcal{L}_{\theta,\phi}(\mathbf{x})$ is known as the negative *variational free energy* because it can also be expressed as a negative "energy" $\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[\log p_\theta(\mathbf{x},\mathbf{z})\right]$ plus the *entropy* of $q_\phi(\mathbf{z}|\mathbf{x})$ given by the negative of the divergence $D_{KL}\left(q_\phi(\mathbf{z}|\mathbf{x})\,||p_\theta(\mathbf{x},\mathbf{z})\right)$ [45].

The KL divergence $D_{KL}\left(q_\phi(\mathbf{z}|\mathbf{x})\,||p_\theta(\mathbf{x},\mathbf{z})\right)$ determines two 'distances':

1. by definition, the KL divergence of the approximate posterior from the true posterior;

2. the gap between the ELBO $\mathcal{L}_{\theta,\phi}(\mathbf{x})$ and the marginal likelihood $\log p_\theta(\mathbf{x})$; this is also called the tightness of the bound. The better $q_\phi(\mathbf{z}|\mathbf{x})$ approximates the true (posterior) distribution $p_\theta(\mathbf{z}|\mathbf{x})$, in terms of the KL divergence, the smaller the gap.

Consequently, that maximization of the ELBO $\mathcal{L}_{\theta,\phi}(\mathbf{x})$ w.r.t. the parameters $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$, will approximately maximize the marginal likelihood $p_\theta(\mathbf{x})$ and minimize the KL divergence of the approximation $q_\phi(\mathbf{z}|\mathbf{x})$ from the true posterior $p_\theta(\mathbf{z}|\mathbf{x})$.

An important property of the ELBO, is that it allows joint optimization w.r.t. all parameters ($\boldsymbol{\theta}$ and $\boldsymbol{\phi}$).

Variational Auto Encoders (VAEs) - Testing Phase: the parameters $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ are not updated so that the encoder and the decoder are the ones learned during training. In this phase, a way of measuring the similarity between the observation and prediction is related to the reconstruction error which gives the anomaly measurement $db0$ (refer to Fig. 5.8) computed as follows:

$$db0 = \left((\boldsymbol{\mu}_x - \widehat{\boldsymbol{\mu}})^T \mathbf{C}^{-1}_{\widehat{\sigma}^2}(\boldsymbol{\mu}_x - \widehat{\boldsymbol{\mu}})\right)^8 \tag{5.12}$$

where $\boldsymbol{\mu}_x$ is the mean vector from the input data with dimension d (for the sake of completeness, $\boldsymbol{\sigma}_x$ is the standard deviation vector from the input data), and $\widehat{\boldsymbol{\mu}}$ and $\widehat{\boldsymbol{\sigma}}$ are the mean and standard deviation vectors from the reconstructed data vector with

the same dimension d. These quantities are the output of neural networks whose input is $\mathbf{x}$ and $\mathbf{x}'$, respectively. $\mathbf{C}_{\widehat{\sigma}^2}^{-1}$ is a covariance matrix given by $\text{diag}(\widehat{\sigma}_1^2, \ldots, \widehat{\sigma}_d^2)$.

One advantage of the VAE framework, with respect to ordinary Variational Inference (VI), is that the inference model (also called recognition model) is now a (stochastic) function of the input variables.

Both the inference model and the generative model can be based on a neural-network. Neural networks are a particularly flexible and computationally scalable type of function approximator. In some cases, models are based on neural networks with multiple 'hidden' layers of artificial neurons forming a deep neural network.

VAEs and GANs seem to have complementary properties: while GANs can generate data of high subjective perceptual quality, they tend to lack full support over the data [22], as opposed to likelihood-based generative models. Indeed, VAEs, like other likelihood-based models, generate more dispersed samples, but are better density models in terms of the likelihood criterion. As such, many hybrid models have been proposed to try to represent the best of both models [10, 22, 41].

In chapter 6, the proposed approach, based on *learning deep generative models* for an *AI-based spectrum abnormality detection in CR*, is discussed in details where the data is represented by a *generalized state vector*. Consequently, networks in the investigated generative models consist of one-dimensional (1D) layers such as 1D-convolutions.

Bibliography

[1] N. Abbas, Y. Nasser, and K. El Ahmad. Recent advances on artificial intelligence and learning techniques in cognitive radio networks. *Eurasip Journal on Wireless Communications and Networking*, 2015, 12 2015.

[2] M.A.J. Apps and M. Tsakiris. The free-energy self: A predictive coding account of self-recognition. *Neuroscience & Biobehavioral Reviews*, 41:85 – 97, 2014. Multisensory integration, sensory substitution and visual rehabilitation.

[3] Y. Bengio. Learning deep architectures for ai. *Found. Trends Mach. Learn.*, 2(1):1–127, January 2009.

[4] Y. Bengio and Y. LeCun. *Scaling learning algorithms towards AI*, page 321–360. In: Large-scale kernel machines, Cambridge, MA: MIT Press, 2007.

[5] M. Bkassiny, Y. Li, and S. K. Jayaweera. A survey on machine-learning techniques in cognitive radios. *IEEE Communications Surveys Tutorials*, 15(3):1136–1159, Third 2013.

[6] A. Clark. Whatever next? predictive brains, situated agents, and the future of cognitive science. *Behavioral and Brain Sciences*, 36(3):181–204, 2013.

[7] D. D. Clark, C. Partridge, J. C. Ramming, and J. T. Wroclawski. A knowledge plane for the internet. In *Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, SIGCOMM '03, pages 3–10, New York, NY, USA, 2003. ACM.

[8] A. Damasio. *The Feeling of What Happens: Body and Emotion in the Making of Consciousness*. In: Harvest Books, October 2000.

[9] S. Dobson, R. Sterritt, P. Nixon, and M. Hinchey. Fulfilling the vision of autonomic computing. *Computer*, 43(1):35–41, Jan 2010.

[10] V. Dumoulin, I. Belghazi, B. Poole, O. Mastropietro, A. Lamb, M. Arjovsky, and A. Courville. Adversarially learned inference. 2016.

[11] S. I. Ele and W. A. Adesola. Artificial neuron network implementation of boolean logic gates by perceptron and threshold element as neuron output function. volume 4, pages 637 – 641, 09 2015.

[12] D. Foster. *Generative Deep Learning*. O'Reilly Media, Inc, 1st edition, 2019. Electronic reproduction. Boston, MA: Safari. Available via World Wide Web: https://www.oreilly.com/library/view/generative-deep-learning/9781492041931/ch01.html.

[13] A. Fotopoulou. *Towards a psychodymamic neuroscience*, page 25–48. Otopoulou, A.,Pfaff, D., Conway, M.A. (Eds.), From the Couch to the Lab: Trends in Psychodynamic Neuroscience. Oxford University Press, Oxford, 2012.

[14] K. Friston. Prediction, perception and agency. *International Journal of Psychophysiology*, 83(2):248 – 252, 2012. Predictive information processing in the brain: Principles, neural mechanisms and models.

[15] K. Friston, T. FitzGerald, F. Rigoli, P. Schwartenbeck, J. O'Doherty, and G. Pezzulo. Active inference and learning. *Neuroscience & Biobehavioral Reviews*, 68:862 – 879, 2016.

[16] K. Friston and S. Kiebel. Cortical circuits for perceptual inference. *Neural Networks*, 22(8):1093 – 1104, 2009. Cortical Microcircuits.

[17] K. Friston, B. Sengupta, and G. Auletta. Cognitive dynamics: From attractors to active inference. *Proceedings of the IEEE*, 102(4):427–445, April 2014.

[18] K. Friston, C. Thornton, and A. Clark. Free-energy minimization and the dark-room problem. *Frontiers in Psychology*, 3:130, 2012.

[19] K. J. Friston. A theory of cortical responses. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 360 1456:815–36, 2005.

[20] K. J. Friston. The free-energy principle: a unified brain theory? nat. rev. neurosci. 11, 127-138. *Nature reviews. Neuroscience*, 11:127–38, 02 2010.

[21] I.J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial networks. *ArXiv e-prints*, June 2014.

[22] A. Grover, M. Dhar, and S. Ermon. Flow-gan: Bridging implicit and prescribed learning in generative models. *CoRR*, abs/1705.08868, 2017.

[23] S. Hameroff and R. Penrose. *Consciousness in the Universe an Updates Review of the "Orch Or" Theory: A Foundational Approach*, chapter 14, pages 517–599. In: Biophysics of Consciousness. World Scientific Publishing Co. Pte. Ltd., 09 2016.

[24] S. Haykin. Cognitive radio: brain-empowered wireless communications. *IEEE Journal on Selected Areas in Communications*, 23(2):201–220, Feb 2005.

[25] S. Haykin. New vision for the world of wireless communications enabled with cognition. In *2011 IEEE 22nd International Symposium on Personal, Indoor and Mobile Radio Communications*, pages 2335–2337, Sep. 2011.

[26] S. Haykin. Cognitive dynamic systems: Radar, control, and radio [point of view]. *Proceedings of the IEEE*, 100(7):2095–2103, July 2012.

[27] J. O. Kephart and D. M. Chess. The vision of autonomic computing. *Computer*, 36(1):41–50, Jan 2003.

[28] D. P. Kingma and M. Welling. An Introduction to Variational Autoencoders. *CoRR*, abs/1906.02691, 2019.

[29] S. Kounev, P. Lewis, K. Bellman, N. Bencomo, J. Cámara, A. Diaconescu, L. Esterle, K. Geihs, H. Giese, S. Götz, P. Inverardi, J. Kephart, and A. Zisman. *The Notion of Self-aware Computing*, pages 3–16. In: Self-Aware Computing Systems, 01 2017.

[30] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521:436–44, 05 2015.

[31] T. S. Lee, D. Mumford, R. Romero, and V.A.F. Lamme. The role of the primary visual cortex in higher level vision. *Vision Research*, 38(15):2429 – 2454, 1998.

[32] P. R. Lewis. Self-aware computing systems: From psychology to engineering. In *Design, Automation Test in Europe Conference Exhibition (DATE), 2017*, pages 1044–1049, March 2017.

[33] A. Mino and G. Spanakis. Logan: Generating logos with a generative adversarial neural network conditioned on color. *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 965–970, 2018.

[34] M. Mirza and S. Osindero. Conditional generative adversarial nets. *CoRR*, abs/1411.1784, 2014.

[35] J. Mitola. Cognitive radio: An integrated agent architecture for software defined radio. 2000.

[36] J. Mitola and G. Q. Maguire. Cognitive radio: making software radios more personal. *IEEE Personal Communications*, 6(4):13–18, Aug 1999.

[37] A. Morin. Levels of consciousness and self-awareness: A comparison and integration of various neurocognitive views. *Consciousness and Cognition*, 15(2):358 – 371, 2006.

[38] M. Nazrul, M. N. Ishlam Patoary, C. Tropper, Z. Lin, R. Mcdougal, and W. Lytton. Neuron time warp. volume 2015, 12 2014.

[39] A. Odena, C. Olah, and J. Shlens. Conditional Image Synthesis With Auxiliary Classifier GANs. *arXiv e-prints*, page arXiv:1610.09585, Oct 2016.

[40] A. Razi and K. Friston. The connected brain: Causality, models, and intrinsic dynamics. *IEEE Signal Processing Magazine*, 33:14–35, 05 2016.

[41] M. Rosca, B. Lakshminarayanan, and S. Mohamed. Distribution matching in variational inference, 2018.

[42] S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Pearson Education Limited, London, United Kingdom, 4rd edition, 2014.

[43] R. Salakhutdinov. Learning deep generative models. *Annual Review of Statistics and Its Application*, 2(1):361–385, 2015.

[44] Y. Sun, X. Wang, and X. Tang. Deep learning face representation by joint identification-verification. *CoRR*, abs/1406.4773, 2014.

[45] C. Zhang, J. Bütepage, H. Kjellström, and S. Mandt. Advances in variational inference. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41:2008–2026, 2017.

[46] Y. Zhao and L. Morales-Tirado. Cognitive radio technology: Principles and practice. In *2012 International Conference on Computing, Networking and Communications (ICNC)*, pages 650–654, Jan 2012.

6 AI-based spectrum abnormality detection: system model *

Following the motivation and objectives for the general scheme of the proposed research, introduced in Chapter 1 and described throughout this thesis, the implementation of the methodology for *abnormality detection at the PHY-layer of CR, by using three deep generative models* (C-GAN, AC-GAN, and VAE), is described in details in this chapter. Some related work is firstly introduced. This is **stage 3** of the general cognitive capability diagram in Fig. 6.1 where the cognitive capability is implemented through AI techniques to enhance the SA. As a contribution, AI-based abnormality detection is performed in a multi-signals wideband spectrum with the mmWave technology.

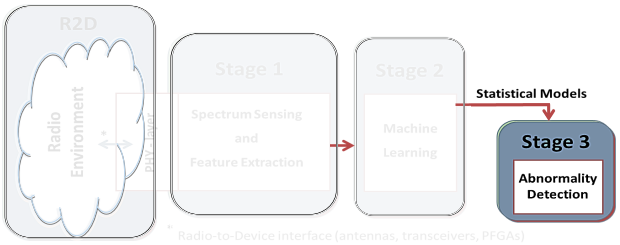


FIGURE 6.1: Abnormality detection phase (stage 3)

6.1 Related work

Recently, it has been emphasized on the implementation of AI techniques in CR in order to accomplish several tasks which are achieved from automatic optimization of parameters and learning of complex models directly from data. A comprehensive overview of learning and reasoning along with details on machine learning techniques for CR are provided in [7, 2], while advanced AI methods in a CR framework can be found in [1]. By using such methodologies and techniques, promising results are obtained in a range of wireless applications. An intelligent modulation and/or coding scheme selection algorithm in CR based on deep reinforcement learning is proposed in [18]. Authors address the imperfect spectrum sensing issue in a cognitive heterogeneous network where secondary transmitters may cause interference to the primary receiver (PR) and make it difficult for the PR to select a proper modulation

* Work submitted to [4] and [5] in *List of publications and under review work*)

and/or coding scheme. Authors in [17] propose a convolutional neural network-based deep learning algorithm for primary user (PU) activity detection in spectrum sensing. The idea is to employ a data-driven deep learning approach, which requires neither a signal-noise probability model nor the PU activity pattern model. Automatic modulation recognition (AMR) in CR through deep learning is investigated in [16]. The authors propose a deep learning-based method combined with two convolutional neural networks (CNNs) to recognize modulation modes. A CNN is trained on samples composed of in-phase and quadrature component signals, while a second CNN is based on constellation diagrams. The objective in ref. [9] is to develop a deep reinforcement learning-based power control method for the problem of spectrum sharing in a CR system consisting of a primary user and a secondary user which work in a non-cooperative manner. CR is also investigated in the TV White Space band and is enabled by machine learning-based spectrum analysis [10].

Abnormality signal detection has also been a significant concern in CR and several techniques based on AI have been implemented to detect abnormal signals. Nevertheless, the literature does not provide exhaustive work in the domain of abnormal signal detection for CR using AI techniques. Therefore, abnormal signal detection in PHY-layer is still a challenging task. A deep predictive coding neural network, trained using image sequences generated from the time-frequency spectrograms and spectral correlation functions of the received RF signal that correspond to normal behaviour, is proposed in [14] where abnormal signals are detected if there is a deviation between actual and predicted image sequence of the signal. On the other hand, in [8], autonomous detection of electromagnetic spectrum anomalies based on spectrum amplitude probability and Hidden Markov Model (HMM) has been proposed. The training process estimates the HMM parameters for different models while the testing process decides which abnormality pattern the testing data belongs to. Rajendran *et al*. proposed in [12] an adversarial autoencoder (AAE) for spectrum anomaly detection with interpretable features based on power spectral density (PSD) data. In [4], authors apply deep-structure auto-encoder neural networks to detect signal anomalies. Signal time-frequency features are used to train the auto-encoders network, which acts as a one-class classifier, relying on the reconstruction error of the network to decide whether the signal is anomalous or not. An interference mitigation algorithm based on neural networks and spectral correlation for wide-band radios, to detect and classify signals with different modulation schemes, is presented in [15]. In ref. [3], classification of RF spectrum modulations and detection of radio frequency anomalies in radar systems is implemented by using CNNs trained on waveform images. The authors proposed two techniques that use the activations of the last hidden layer of the CNNs to detect anomalies. The aforementioned citations played an important role

in implementing AI-based techniques in CR to detect abnormalities inside the radio spectrum. However, in this thesis, a framework that can be implemented at physical layer level and deals with high data dimensionality is proposed, providing by that a framework that mimics the real-world applications and the dynamic nature of CRs.

6.2 Approach based on Dynamic Bayesian Networks

Before investigating the proposed approach based on C-GAN, AC-GAN, and VAE, a previous comparative study with two generative model-based approaches was also conducted by detecting abnormal behaviours inside the radio spectrum in two different applications. Specifically, to investigate the first two functionalities of SA, which are essential to achieve the third functionality, two AI-based Abnormality Detection techniques were proposed, the *Conditional Generative Adversarial Networks* (C-GANs) and the *Dynamic Bayesian Networks* (DBNs). Both of the techniques are based on learning Generative Models (GANs and DBNs are examples of such models) used for probabilistic reasoning on the observed data [6]. GAN is one of the most crucial research avenues in the field of AI as an unsupervised learning technique and its outstanding data generation capacity has received extensive attention [11]. GAN was proposed in the study due to its fast and accurate inferences which are based on a likelihood-free algorithm. DBNs belong to the Switching Dynamic Models (SDMs) that are used successfully to improve decision making and tracking capabilities in different applications [13]. DBN was proposed due to its ability to model linear and non-linear dynamics corresponding to switching variables employing Bayesian filters such as Kalman and Particle filters.

The two applications differed in the data dimensionality and the PHY-layer level at which the AI-method was implemented. In the first application, the AI method (C-GAN) was applied just after the receiving antenna and the down-conversion process where multi-signals representations were extracted from a wideband spectrum with a high sampling rate and, consequently, high dimensionality data. On the other hand, in the second application, the method (DBN) was employed after down-conversion, cyclic prefix removal and Fast Fourier Transform (FFT) block at the CR device side where signals with low dimensionality and low sampling rate were extracted.

By implementing the DBN, it was possible to learn switching models from data series of generalized states where each switching variable could be associated with a different linear dynamic model. Therefore, such an approach was applied to low dimensionality data (Second Application), where the number of possible switching dynamic models to be included in different Bayesian (e.g. Kalman) filters was limited.

On the contrary, C-GAN was employed for high-dimensionality data (First Application) because learning a DBN in this case with high dimensionality data would have generated a vast vocabulary of switching variables, making the model computationally intractable. GANs can effectively manage a high number of different dynamic models implicitly but (and this is their main drawback) they are unable to manage uncertainty as a DBN does with probabilistic knowledge.

However, the work in this thesis investigates *both GAN-based models and VAE to perform abnormality detection*.

6.3 The proposed approach based on Deep Generative Model

The general scheme of the proposed research is depicted in Fig. 6.2. In this section, a clear and complete overview of the contributions and innovations, provided by this thesis, are represented in the blocks of the general scheme. It can be employed in the *Cognitive PHY-layer security with mmWave transmission*. The *radio environment* represents wireless communication in which transmissions are involved in the mmWave band. Motivation for mmWave is given in Sec. 7.1. A CR system observes and gathers information about the spectrum occupancy where multiple signals dynamically occupy the available channels. However, processing and sensing such a dynamic spectrum in the considered scenario requires suitable techniques. To this end, *Stockwell Transform (ST)* is used to extract the time-frequency representation of the spectrum following the approach presented in Sec. 3.5.2. From such a representation, a *generalized state vector* is formed, as defined in ref. [5]. It consists of the current state in terms of amplitude (A) and its first-order derivative ($\dot{A}$):

$$\mathbf{x} = [A_{ch,k},\ \dot{A}_{ch,k}]; \quad \text{ch} \in \{1, \ldots, N\} \tag{6.1}$$

where k is the time instant at which each value A related to the ch-th channel is extracted from ST and N is the total number of channels. Alternatively, different features from the ST representation can also be analysed as mentioned in Sec. 3.5.4. From the

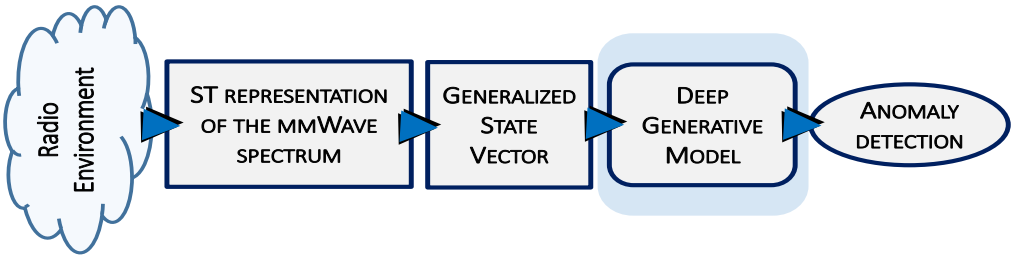


FIGURE 6.2: The deep model-based abnormality detection scheme for CR

generalized state vector, the proposed *deep generative models* presented in this chapter will learn the dynamics of the radio environment and how the signals are evolving with time. To this end, one-dimensional (1D) layers, such as 1D-convolutions, are employed in these models (Secs. 6.3.2 - 6.3.3). The *anomaly detection* exploits indicators, denoted *db*0, from the generative models, as discussed in Sec. 5.6.

For validation purpose, a wideband spectrum at 28 GHz with 800 MHz total frequency range consisting of 8 channels has been generated by using the mmWave system in Sec. 2.3.3. However, as mentioned in Sec. 2.2.3, several mmWave bands have been allocated in different parts of the mmWave spectrum, from the 28 GHz band to the 80 GHz band, even with larger bandwidths than 800 MHz. Considering that, it is worth mentioning that the complexity of the proposed abnormality detection approach, in Fig. 6.2, is mostly related to the complexity of the ST computation (related to the length of the input time-series) and the complexity of the deep generative model (related to the length of the generalized state vector). These two factors depend on the sampling rate of the signal, on the frequency range of the observed spectrum and, from a certain point of view, on the number of channels inside it.

To reduce the complexity of the algorithm when the frequency range (relative to the specific band of interest) is large, the ST can be computed on a reduced number of frequency samples (whenever all the frequency information inside each channel is not necessary), and/or the algorithm can analyse only the portion of the spectrum of interest (whenever the dynamic spectrum is restricted to a limited portion of the whole frequency band). Alternatively, abnormality detection can be performed by sequentially shifting a frequency window, with shorter range, in the mmWave band; for example, in a centralized network where the coordinator node can be a base station. Moreover, compressed sampling can also be applied to the input time-series.

In any case, under the same frequency range, the abnormality detection performance would not be influenced by the mmWave band. Indeed, the learning method is applied to features (forming the generalized state vector) extracted from the observed wideband signal in the specific mmWave band; an example is described in Sec. 6.3.1. These features would not depend much on the specific band of the spectrum in the mmWave where they are extracted.

6.3.1 Feature extraction through ST and generalized state vector

An example of raw ST representation of a mmWave signal (relative to the 28 GHz band) in a dynamic scenario is shown in Fig. 6.3 (left image) where a fixed signal is found at channel number 4 whereas, in the meantime, a moving signal jumps in the spectrum at different time instants (sequentially, ch 7, ch 2, and ch 6). The delay index is the shift number of the sliding window of the ST. Signals can be detected from the

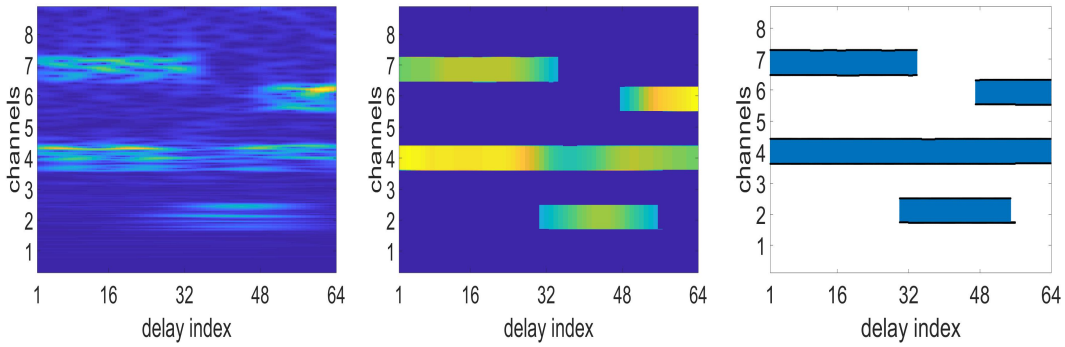


FIGURE 6.3: ST representation of a wideband dynamic signal: raw representation (left), occupied areas (centre) and occupancy represented by blue areas with contours represented as black lines (right)

ST representation to obtain occupied areas in both the frequency and time domains, as shown in Fig. 6.3 (middle image). Afterwards, occupancy (considered as energy locations) and corresponding contours for each occupied area can be extracted as in Fig. 6.3 (right image). A detailed description of the theoretical aspects of the ST employed in this work (with description of the proposed approach for high sampling rate signals) can be found in Sec. 3.5.2. This kind of analysis allows for the extraction of features such as bandwidth, central frequency, received power or amplitude, and variance. The extracted values can be mapped into the time-frequency location of the corresponding signals and, then, represented as bi-dimensional localized dynamic features. Derivatives can be computed to obtain the generalized state vector for each of the considered features. In this case, since the generalized state vector is extracted from columns from the ST representation and the corresponding derivatives, its dimensionality is high and mostly related to the frequency resolution in the ST. The input data to the C-GAN/AC-GAN or VAE based on this kind of representation is described in detail in Chapter 7, where the amplitude is the feature extracted and used for validation purposes.

In the following, detailed algorithm descriptions for the three generative models are presented.

6.3.2 Deep Generative Model Block: C-GAN/AC-GAN

A detailed description of the algorithm implementation is provided below.

Data generation phase

These steps allows data generation from the wideband signal: *i)* frequency band extraction and down-sampling; *ii)* ST representation of the wideband spectrum; *iii)* feature extraction and representation on a 2D basis.

Training phase

The C-GAN/AC-GAN is trained on normality data samples (let L be the training length) related to ST representation of the feature(s) according to the following steps:

- set gradients of all G and D model parameters to zero;
- compute d_real_loss and d_fake_loss through the adversarial loss function (and auxiliary loss function);
- perform the back-propagation of the $d\_loss = (d\_real\_loss + d\_fake\_loss)/2$ by computing gradient of the losses w.r.t all the parameters in the losses;
- perform a parameter update based on the current gradients.

When training finishes, the learnt C-GAN/AC-GAN model is ready to be used in the testing process.

Testing phase

Testing is performed according to the following steps (let T the testing length):

- the G and D optimizers are switched off so that the C-GAN/AC-GAN model is the one learnt during training;
- apply C-GAN/AC-GAN to normality and abnormality data samples related to ST representation of the feature(s);
- compute the d_real_loss, d_fake_loss, and d_loss through the adversarial loss function (and auxiliary loss function);
- compute the abnormality indicator $db0$ from the $|d\_real\_loss - d\_fake\_loss|$ which will be used for the binary classification at the time instant t (t-th sample).

Further details can be found in the following lines of Python code. Note that AC-GAN includes one-dimensional (1D) layers such as 1D-convolutions.

C-GAN:

```
#  ----------- G E N E R A T O R ------------------
class Generator(nn.Module):
    def __init__(self):
        super(Generator, self).__init__()

        self.label_emb = nn.Embedding(opt.n_classes, opt.
            n_classes)
```

```
        def block(in_feat, out_feat, normalize=True):
            layers = [nn.Linear(in_feat, out_feat)]
            if normalize:
                layers.append(nn.BatchNorm1d(out_feat, 0.8))
            layers.append(nn.LeakyReLU(0.2, inplace=True))
            return layers

        self.model = nn.Sequential(
            *block(opt.latent_dim+opt.n_classes, 128, normalize=
                False),
            *block(128, 256),
            *block(256, 512),
            *block(512, 1024),
            nn.Linear(1024, int(np.prod(img_shape))),
            nn.Tanh()
        )

    def forward(self, noise, labels):
        gen_input = torch.cat((self.label_emb(labels), noise),
            -1)
        img = self.model(gen_input)
        img = img.view(img.size(0), *img_shape)
        return img

# ------- D I S C R I M I N A T O R ---------
class Discriminator(nn.Module):
    def __init__(self):
        super(Discriminator, self).__init__()

        self.label_embedding = nn.Embedding(opt.n_classes, opt.
            n_classes)

        self.model = nn.Sequential(
            nn.Linear(opt.n_classes + int(np.prod(img_shape)),
                512),
            nn.LeakyReLU(0.2, inplace=True),
            nn.Linear(512, 512),
            nn.Dropout(0.4),
            nn.LeakyReLU(0.2, inplace=True),
            nn.Linear(512, 512),
            nn.Dropout(0.4),
            nn.LeakyReLU(0.2, inplace=True),
            nn.Linear(512, 1)
        )

    def forward(self, img, labels):
```

```
        d_in = torch.cat((img.view(img.size(0), -1), self.
            label_embedding(labels)), -1)
        validity = self.model(d_in)
        return validity

# Loss function
adversarial_loss = torch.nn.MSELoss()
# Initialize generator and discriminator
generator = Generator()
discriminator = Discriminator()
# Optimizers
optimizer_G = torch.optim.Adam(generator.parameters(), lr=opt.lr
    , betas=(opt.b1, opt.b2))
optimizer_D = torch.optim.Adam(discriminator.parameters(), lr=
    opt.lr, betas=(opt.b1, opt.b2))
optimizer_G.zero_grad(), optimizer_D.zero_grad()  # only
    training
# Generate a batch of vectors
gen_vects = generator(z, gen_labels)
# Generator's ability to fool the discriminator
validity = discriminator(gen_vects, gen_labels)
g_loss = adversarial_loss(validity, valid)
# Discriminator's ability to discriminate: Loss for real vectors
validity_real = discriminator(GSV, labels)
d_real_loss = adversarial_loss(validity_real, valid)
# Discriminator's ability to discriminate: Loss for fake vectors
validity_fake = discriminator(gen_vects.detach(), gen_labels)
d_fake_loss = adversarial_loss(validity_fake, fake)
# Loss
d_loss = (d_real_loss + d_fake_loss) / 2
d_loss.backward()  # (only training phase)
# Abnormality indicator  # (only testing phase)
Dloss[i] = torch.abs(d_real_loss - d_fake_loss)
db0[i] = np.percentile(Dloss[i-300+1:i+1].detach().numpy(),97)
#  ============================================================
```

AC-GAN:

```
#  --------------- G  E  N  E  R  A  T  O  R ------------------
class Generator(nn.Module):
    def __init__(self):
        super(Generator, self).__init__()
        self.label_emb = nn.Embedding(opt.n_classes, opt.
            latent_dim)
        self.init_size = opt.img_size // 4  # Initial size
            before upsampling
```

```
            self.l1 = nn.Sequential(nn.Linear(opt.latent_dim, 128 *
                (self.init_size) ** 1))

            self.conv_blocks = nn.Sequential(
                nn.BatchNorm1d(128),
                nn.Upsample(scale_factor=2),
                nn.Conv1d(128, 128, 3, stride=1, padding=1),
                nn.BatchNorm1d(128, 0.8),
                nn.LeakyReLU(0.2, inplace=True),
                nn.Upsample(scale_factor=2),
                nn.Conv1d(128, 64, 3, stride=1, padding=1),
                nn.BatchNorm1d(64, 0.8),
                nn.LeakyReLU(0.2, inplace=True),
                nn.Conv1d(64, opt.channels, 3, stride=1, padding=1),
                nn.Tanh(),
            )

        def forward(self, noise, labels):
            gen_input = torch.mul(self.label_emb(labels), noise)
            out = self.l1(gen_input)
            out = out.view(out.shape[0], 128, self.init_size)
            img = self.conv_blocks(out)
            return img

    #  ------- D  I  S  C  R  I  M  I  N  A  T  O  R ---------
    class Discriminator(nn.Module):
        def __init__(self):
            super(Discriminator, self).__init__()
            def discriminator_block(in_filters, out_filters, bn=True
                ):
                block = [nn.Conv1d(in_filters, out_filters, 3, 2, 1)
                    , nn.LeakyReLU(0.2, inplace=True), nn.Dropout
                    (0.25)]
                if bn:
                    block.append(nn.BatchNorm1d(out_filters, 0.8))
                return block

            self.conv_blocks = nn.Sequential(
                *discriminator_block(opt.channels, 16, bn=False),
                *discriminator_block(16, 32),
                *discriminator_block(32, 64),
                *discriminator_block(64, 128),
            )

            # The height and width
            ds_size = opt.img_size // 2 ** 2
```

```
        # Output layers
        self.adv_layer = nn.Sequential(nn.Linear(128 * ds_size
            //4 ** 1, 1), nn.Sigmoid())
        self.aux_layer = nn.Sequential(nn.Linear(128 * ds_size
            //4 ** 1, opt.n_classes), nn.Softmax())

    def forward(self, img):
        out = self.conv_blocks(img)
        out = out.view(out.shape[0], -1)
        validity = self.adv_layer(out)
        label = self.aux_layer(out)
        return validity, label

# Loss functions
#adversarial_loss = L8Loss()
auxiliary_loss = torch.nn.CrossEntropyLoss()
# Initialize generator and discriminator
generator = Generator()
discriminator = Discriminator()
# Optimizers
optimizer_G = torch.optim.Adam(generator.parameters(), lr=opt.lr
    , betas=(opt.b1, opt.b2))
optimizer_D = torch.optim.Adam(discriminator.parameters(), lr=
    opt.lr, betas=(opt.b1, opt.b2))
optimizer_G.zero_grad(), optimizer_D.zero_grad() # only training
# Generate a batch of vectors
gen_vects = generator(z, gen_labels)
# Loss measures generator's ability to fool the discriminator
validity, pred_label = discriminator(gen_vects)
g_loss = 0.5 * (torch.mean((validity - valid)**8) +
    auxiliary_loss(pred_label, gen_labels))
# Discriminator's ability to discriminate: Loss for real vectors
real_pred, real_aux = discriminator(GSV)
d_real_loss = (torch.mean((real_pred - valid)**8) +
    auxiliary_loss(real_aux, labels)) / 2
# Discriminator's ability to discriminate: Loss for fake vectors
fake_pred, fake_aux = discriminator(gen_vects.detach())
d_fake_loss = (torch.mean((fake_pred - fake)**8) +
    auxiliary_loss(fake_aux, gen_labels.long())) / 2
# Loss
d_loss = (d_real_loss + d_fake_loss) / 2
d_loss.backward()  # (only training phase)
# Abnormality indicator  # (only testing phase)
Dloss[i] = torch.abs(d_real_loss - d_fake_loss)
db0[i] = np.percentile(Dloss[i-300+1:i+1].detach().numpy(),98)
#  ============================================================
```

6.3.3 Deep Generative Model Block: VAE

The **Data generation phase** is the same as in C-GAN/AC-GAN.

Training phase

The VAE is trained on normality data samples according to the following steps:

- set gradients of all parameters to zero;
- compute mean and variance of the input vector, the encoded signal, and the reconstructed vector of the data;
- compute the loss from L^p and KL;
- perform the back-propagation of the loss by computing the gradient of the loss w.r.t. all the parameters;
- update the networks' parameters based on the current gradient.

When training finishes, the learnt VAE model is ready to be used in the testing process.

Testing phase

Testing is performed according to the following steps:

- the optimizer is not active so that the VAE model is the one learnt during training;
- apply VAE to normality and abnormality data from ST representation;
- compute the loss from L^p and KL;
- compute the abnormality indicator $db0$ which will be used for the binary classification at time instant t (t-th sample).

Further details can be found in the following lines of Python code. Note that VAE includes one-dimensional (1D) layers such as 1D-convolutions.

VAE:

```
#  ---------- E N C O D E R - D E C O D E R -----------------
class VAE(nn.Module):
    def __init__(self, nz):
        super(VAE, self).__init__()
        self.have_cuda = True
        self.nz = nz
```

```
        self.encoder = nn.Sequential(
            nn.Conv1d(nc, ndf, 4, 4, 1, bias=False),
            nn.LeakyReLU(0.2, inplace=True),
            nn.Conv1d(ndf, ndf * 2, 4, 4, 1, bias=False),
            nn.BatchNorm1d(ndf * 2, eps=100000e-05, momentum
                =0.05),
            nn.LeakyReLU(0.2, inplace=True),
            nn.Conv1d(ndf * 2, ndf * 4, 4, 4, 1, bias=False),
            nn.BatchNorm1d(ndf * 4, eps=100000e-05, momentum
                =0.05),
            nn.LeakyReLU(0.2, inplace=True),
            nn.Conv1d(ndf * 4, 1024, 4, 4, 0, bias=False),
            nn.LeakyReLU(0.2, inplace=True),
        )
        self.decoder = nn.Sequential(
            nn.ConvTranspose1d(1024, ngf * 8, 3, 3, 0, bias=
                False),
            nn.BatchNorm1d(ngf * 8, eps=1e-05, momentum=0.05),
            nn.ReLU(True),
            nn.ConvTranspose1d(ngf * 8, ngf * 4, 6, 5, 1, bias=
                False),
            nn.BatchNorm1d(ngf * 4, eps=1e-05, momentum=0.05),
            nn.ReLU(True),
            nn.ConvTranspose1d(ngf * 4, ngf * 2, 13, 3, 1, bias=
                False),
            nn.BatchNorm1d(ngf * 2, eps=1e-05, momentum=0.05),
            nn.ReLU(True),
            nn.ConvTranspose1d(ngf * 2,     nc, 13, 5, 1, bias=
                False),
            nn.Sigmoid()
        )
        #
        self.fc1 = nn.Linear(1024, 512)
        self.fc21 = nn.Linear(512, nz)
        self.fc22 = nn.Linear(512, nz)
        self.fc3 = nn.Linear(nz, 512)
        self.fc4 = nn.Linear(512, 1024)
        self.fc31 = nn.Linear(256, 256)
        self.fc32 = nn.Linear(256, 256)
        self.lrelu = nn.LeakyReLU()
        self.relu = nn.ReLU()

    def encode(self, x):
        mu_x = self.fc31(x)
        logvar_x = self.fc32(x)
        self.encoder = self.encoder.type(FloatTensor)
```

```
        conv = self.encoder(x.type(FloatTensor))
        h1 = self.fc1(conv.view(-1, 1024))
        return self.fc21(h1), self.fc22(h1), mu_x, logvar_x

    def decode(self, z):
        h3 = self.relu(self.fc3(z))
        deconv_input = self.fc4(h3)
        deconv_input = deconv_input.view(-1,1024,1)
        self.decoder = self.decoder.type(FloatTensor)
        mu_hat = self.fc31(self.decoder(deconv_input.type(
            FloatTensor)))
        logvar_hat = self.fc32(self.decoder(deconv_input.type(
            FloatTensor)))
        return self.decoder(deconv_input.type(FloatTensor)),
            mu_hat, logvar_hat

    def reparametrize(self, mu, logvar):
        std = logvar.mul(0.5).exp_()
        eps = torch.cuda.FloatTensor(std.size()).normal_()
        eps = Variable(eps)
        return eps.mul(std).add_(mu)

    def forward(self, x):
        self = self.type(FloatTensor)
        mu, logvar, mu_x, logvar_x = self.encode(x.type(
            FloatTensor))
        z = self.reparametrize(mu, logvar)
        decoded, mu_hat, logvar_hat = self.decode(z)
        return decoded, mu, logvar, mu_hat, logvar_hat, mu_x,
            logvar_x

def L8Loss(recon_x, x, mu, logvar):
    loss = torch.mean((recon_x - x)**8)
    KLD = -0.5 * torch.sum(1 + logvar - mu.pow(2) - logvar.exp()
        )
    return 1*loss + 125*loss * KLD

# MODEL and OPTIMIZER:
model = vae_conv_model.VAE(2048)
optimizer = torch.optim.Adam(model.parameters(), lr=0.0000001,
    betas=(0.005, 0.65), eps=10e-01, weight_decay=0, amsgrad=
    False)
# TRAINING phase:
optimizer.zero_grad()
recon_batch, mu, logvar, mu_hat, logvar_hat, mu_x, logvar_x =
    model(data)
```

```
loss = L8Loss(recon_batch, data, mu, logvar)
loss.backward()
optimizer.step()
# TESTING phase:
recon_batch, mu, logvar, mu_hat, logvar_hat, mu_x, logvar_x =
    model(data)
var_hat = logvar_hat.exp_()
D = mu_x.cpu().detach().numpy()
M = mu_hat.cpu().detach().numpy()
V = var_hat.cpu().detach().numpy()
x_mu = matrix(D[0,0,0:256] - M[0,0,0:256])
inv = np.linalg.inv(np.diag(V[0,0,0:256]))
# Abnormality indicator
abn[i] = math.pow(x_mu.T * inv * x_mu,8)
db0[i] = np.percentile(abn[i-700+1:i+1].detach().numpy(),98)
#  ============================================================
```

Finally, binary classification is performed at each data sample (instantaneous generalized state vector) by comparing the value of the abnormality indicator, as in Eqs. 5.3 and 5.7 for C-GAN and AC-GAN, respectively, and in Eq. 5.12 for VAE, at that time instant with a range of thresholds to differentiate normal samples from abnormal samples and, consequently, ROC curves are generated.

Further information is provided in Chapter 7.

6.3.4 Abnormality detection and performance evaluation

Distance metrics are used to provide an abnormality indicator ($db0$) as described in Sec. 5.6, to decide weather the signals inside the radio spectrum follow a normal activity or not. In order to evaluate the performance of the proposed methods, a range of confidence thresholds have been used to build a corresponding Receiver Operating Characteristic (ROC) curve. These confidence thresholds have then been applied to the abnormality signals provided in the testing phase, which are denoted as $db0$. The ROC curve represents the probability of detection, P_d, over the probability of false alarm, P_{fa}. Where, P_d is the number of times when abnormalities are correctly identified, and P_{fa} refers to the times where abnormalities are classified incorrectly (when normal behaviour is identified as abnormal behaviours). Experiments are discussed in Chapter 7.

Bibliography

[1] N. Abbas, Y. Nasser, and K. E. Ahmad. Recent advances on artificial intelligence and learning techniques in cognitive radio networks. *EURASIP Journal on Wireless Communications and Networking*, 2015(1):174, Jun 2015.

[2] C. Clancy, J. Hecker, E. Stuntebeck, and T. O'Shea. Applications of machine learning to cognitive radio networks. *IEEE Wireless Communications*, 14(4):47–52, August 2007.

[3] M. A. Conn and D. Josyula. Radio frequency classification and anomaly detection using convolutional neural networks. In *2019 IEEE Radar Conference (RadarConf), Boston, MA, USA*, pages 1–6, April 2019.

[4] Q. Feng, Y. Zhang, C. Li, Z. Dou, and J. Wang. Anomaly detection of spectrum in wireless communication via deep auto-encoders. *The Journal of Supercomputing*, 73(7):3161–3178, Jul 2017.

[5] K. Friston, B. Sengupta, and G. Auletta. Cognitive dynamics: From attractors to active inference. *Proceedings of the IEEE*, 102(4):427–445, April 2014.

[6] X. Gao, Z.-Y. Zhang, and L.-M. Duan. A quantum machine learning algorithm based on generative models. *Science Advances*, 4(12), 2018.

[7] L. Gavrilovska, V. Atanasovski, I. Macaluso, and L. A. DaSilva. Learning and reasoning in cognitive radio networks. *IEEE Communications Surveys Tutorials*, 15(4):1761–1777, Fourth 2013.

[8] W. Honghao, J. Yunfeng, and W. Lei. Spectrum anomalies autonomous detection in cognitive radio using hidden markov models. In *2015 IEEE Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, pages 388–392, Dec 2015.

[9] X. Li, J. Fang, W. Cheng, H. Duan, Z. Chen, and H. Li. Intelligent power control for spectrum sharing in cognitive radios: A deep reinforcement learning approach. *IEEE Access*, 6:25463–25473, 2018.

[10] Y. Ma, Y. Gao, C. Fu, W. Rong, Z. Xiong, and S. Cui. TV White Space Spectrum Analysis based on Machine Learning. *Journal of Communications and Information Networks*, 4(2):68–80, June 2019.

[11] Z. Pan, W. Yu, X. Yi, A. Khan, F. Yuan, and Y. Zheng. Recent progress on generative adversarial networks (gans): A survey. *IEEE Access*, 7:36322–36333, 2019.

[12] S. Rajendran, W. Meert, V. Lenders, and S. Pollin. Unsupervised wireless spectrum anomaly detection with interpretable features. *IEEE Transactions on Cognitive Communications and Networking*, 5(3):637–647, 09 2019.

[13] M. Ravanbakhsh, M. Baydoun, D. Campo, P. Marin, D. Martin, L. Marcenaro, and C. S. Regazzoni. Learning multi-modal self-awareness models for autonomous vehicles from human driving. In *2018 21st International Conference on Information Fusion (FUSION)*, pages 1866–1873, July 2018.

[14] N. Tandiya, A. Jauhar, V. Marojevic, and J. H. Reed. Deep predictive coding neural network for rf anomaly detection in wireless networks. In *2018 IEEE International Conference on Communications Workshops (ICC Workshops)*, pages 1–6, May 2018.

[15] A. Toma, T. Nawaz, Y. Gao, L. Marcenaro, and C. S. Regazzoni. Interference mitigation in wideband radios using spectrum correlation and neural network. *IET Communications*, 13(10):1336–1347, 2019.

[16] Y. Wang, M. Liu, J. Yang, and G. Gui. Data-driven deep learning for automatic modulation recognition in cognitive radios. *IEEE Transactions on Vehicular Technology*, 68(4):4074–4077, April 2019.

[17] J. Xie, C. Liu, Y. Liang, and J. Fang. Activity pattern aware spectrum sensing: A cnn-based deep learning approach. *IEEE Communications Letters*, 23(6):1025–1028, June 2019.

[18] L. Zhang, J. Tan, Y. Liang, G. Feng, and D. Niyato. Deep reinforcement learning-based modulation and coding scheme selection in cognitive heterogeneous networks. *IEEE Transactions on Wireless Communications*, 18(6):3281–3294, June 2019.

7 AI-based spectrum abnormality detection: experimental results *

The experiments described in this chapter have been performed to show and validate the effective feasibility of the proposed approach (represented in Fig. 6.2 and described in Chapter 6) to detect *abnormal behaviours* (Fig. 7.1) in the 28 GHz mmWave spectrum band for a data-driven self-awareness. After training the C-GAN, AC-GAN, and VAE on a generalized state vector, performance of these three generative models, when used as abnormality detector, are presented and compared throughout this chapter.

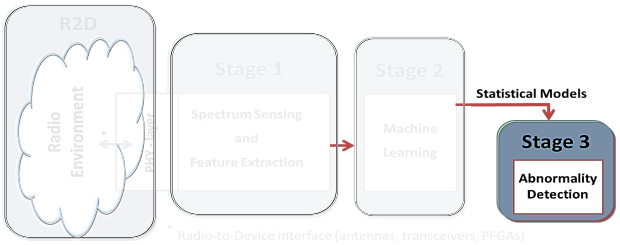


FIGURE 7.1: Abnormality detection phase (stage 3)

7.1 Introduction

Millimeter-wave is proposed to address the spectrum scarcity issue and increase the radio spectrum utilization [8]. The mmWave provides sizeable available bandwidth at high frequencies which operate in the range of 30 to 300 GHz, offering low latency and high-speed data connection [11, 1]. Such frequencies impose several limitations due to the fact that the signal will suffer from high propagation loss and get distorted due to raindrops and humidity absorption as well as its sensitivity to blockages, making the implementation of the mmWave communications preferable in applications with small cells and heterogeneous networks which are efficient to serve the Internet of Things (IoT) and Vehicle to Everything (V2X) [9, 10]. Besides, the fifth-generation (5G) technology will provide a system structure for these emerging V2X and IoT applications that require high reliability and strict delay for secure message delivery between transmitters and receivers which impose the need of an efficient hybrid access scheme for licensed and unlicensed spectrum in mmWave bands. Thus, CR has

* Work submitted to [4] and [5] in *List of publications and under review work*)

been proposed to manage the dynamic spectrum access in mmWave communications [3].

Obviously, precise detection of mmWave spectrum anomalies is crucial to enhance the physical layer security and improve the system's performance.

Concerning the contribution and innovation, in the proposed framework for spectrum anomaly detection, three deep generative models are compared: the *Conditional Generative Adversarial Network* (C-GAN), the *Auxiliary Classifier GAN* (AC-GAN) and the *Variational Auto Encoder* (VAE). These generative models are investigated and employed in the *mmWave communications* enabled by CR to learn a representation of the dynamic spectrum following probabilistic reasoning. In this way, *data-driven self-awareness* can be enabled for spectrum security. Specifically, a generalized state vector, consisting of the signal feature (amplitude) extracted from the Stockwell Transform (ST) and the corresponding derivatives, is formed and used to construct the network that, consequently, detects any abnormal signals related to abnormal behaviours inside the 28 GHz mmWave band. However, other frequency bands in the mmWave can be considered in the same way.

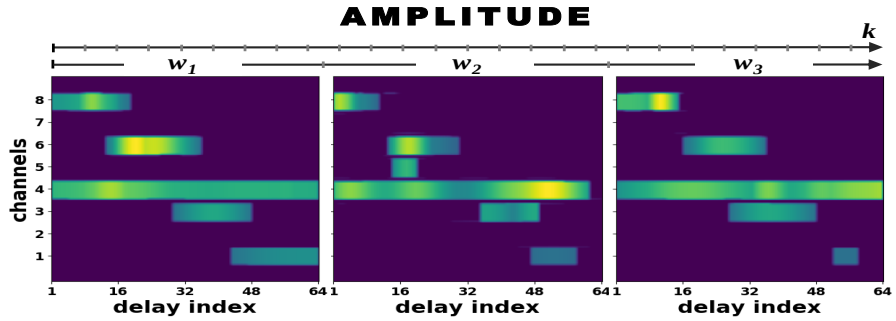
Moreover, by using the NI mmWave testbed described in Sec. 2.3.3, a real dataset has been generated for the purpose of the experiments presented in this chapter. The whole mmWave dataset is divided into two sets: one for the training phase which represents the normal behaviour (no malicious behaviour) of the signals inside the spectrum and the second is used during the testing phase including four different anomaly modalities in which the behaviour of the signal is different from the normal one.

Firstly, results when the generative model is a C-GAN are presented in Secs. 7.2-7.4. While, in Sec. 7.5, the results refer to AC-GAN and VAE.

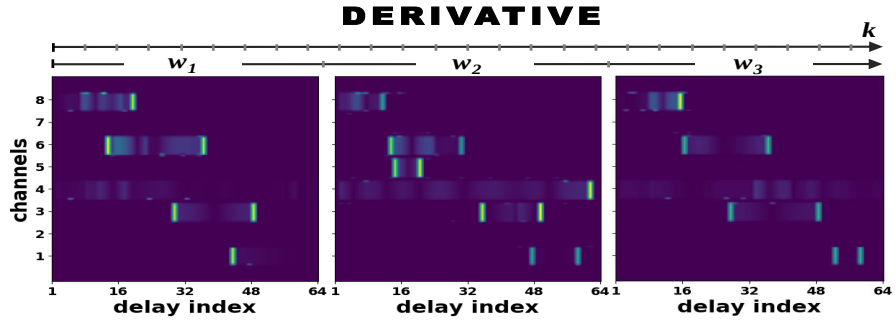
The Adam optimizer is used to train G and D of C-GAN and AC-GAN as well as the encoder and decoder of VAE. MSE loss is used as adversarial loss in C-GAN, while L^p loss (with p = 8) in AC-GAN which also includes a Cross-Entropy loss as an auxiliary loss. The KL divergence is included in the loss function in VAE. Experiments have been performed on 'NVIDIA® GeForce® GTX 1080 Ti' GPU.

7.2 Training of the C-GAN model

A large number of data samples of the bi-dimensional localized dynamic feature from ST representation with a constant pattern is obtained from mmWave wideband spectrum observations (discussed in Sec. 2.3.3) to validate the C-GAN based method. During the training phase, thousands of dynamic feature samples, which represent the normality pattern, are used to train the C-GAN model where the amplitude is



(a) Amplitude



(b) Derivative

FIGURE 7.2: Independent data samples from bi-dimensional localized dynamic amplitude. Pattern used as normality data (in this example: $w = 1, 2, 3$; 64 shifts in the ST; and $k = 1, \cdots, 192$)

considered as a feature. Three samples are shown in Fig. 7.2(a) for better understanding of the dataset; in this case, the ST (with 64-time shifts of the sliding window denoted as delay index) is sequentially applied to consecutive time windows (indexed by $w = 1, 2, 3, \ldots$) by exploiting the approach in Sec. 3.5.2. The k axis represents the time domain in terms of 64 shifts, while the vertical axis represents the frequency domain consisting of 8 channels divided into 128 sub-channels.
Specifically, the ***normality data*** for the training phase shows the following pattern:

- Normality pattern: a fixed signal is occupying *ch-4* and a moving signal sequentially jumps between *ch-8*, *ch-6*, *ch-3*, *ch-1*.

In the normality data, no malicious behaviour is present, but there is a fixed (admissible) signal and a second signal with a legitimate behaviour or strategy. Without loss of generality, a separation between consecutive channels has been introduced to clearly distinguish the different signals in the observed spectrum. The performance of C-GAN is evaluated by using two different scenarios related to two kinds of input data.

- **Scenario I**: the input data, consisting of $L = 33600$ images (some of them are illustrated in Fig. 7.2(a)), is applied as an input to the C-GAN following

the most popular usage of GAN-based methods. In this case, each image is represented as a matrix $\mathbf{X}_w$ consisting of the amplitude (the feature extracted) values related to each time instant (columns) and channel (rows).

- **Scenario II**: the input data ($L = 26240$ samples) is a generalized state vector which is sequentially given the values relative to each column from samples like the ones shown in Fig. 7.2(a) and concatenated with the corresponding derivative vector, namely columns from samples like in Fig. 7.2(b) (for the sake of clarity, absolute values of the derivatives are shown in the picture). The state vector is thus composed of 256 elements at each time instant k. Specifically, let $\mathbf{x}_k$ denote the generalized state vector, where k represents the discrete-time at which it is extracted.
  This would be the first time in which the generalized state vector is investigated as input data to GAN-based models.

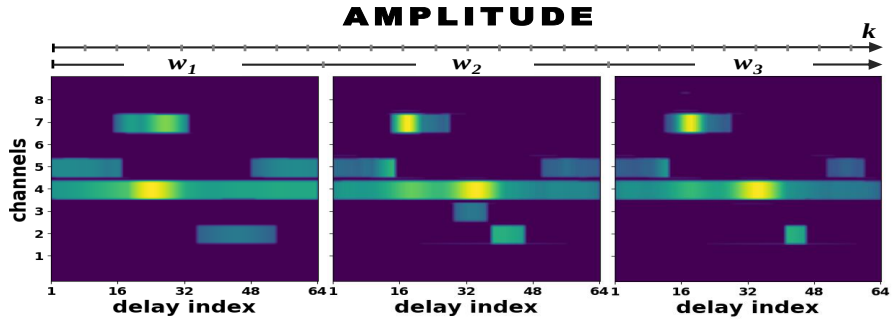
By providing the normality data, the G model and the D model are learnt in an unsupervised way during the training phase. Indeed, in this work, the conditioning information, the vector $\mathbf{y}$ in Fig. 5.7(a), consists of a fictitious input label because it is assigned the same value regardless of the input data $\mathbf{x}$ whether normal or abnormal. This choice allows us to employ and investigate the C-GAN as an unsupervised learning model. The training data is used to train neurons in the latent space of the generator network G; note that it has been shown that in generative models each neuron learns to detect specific types of features from the input data [7]. Intrinsic clustering on input data is also obtained thanks to neurons that learn to detect similarity characteristics of groups of input samples.

7.3 C-GAN as abnormal behaviour detector

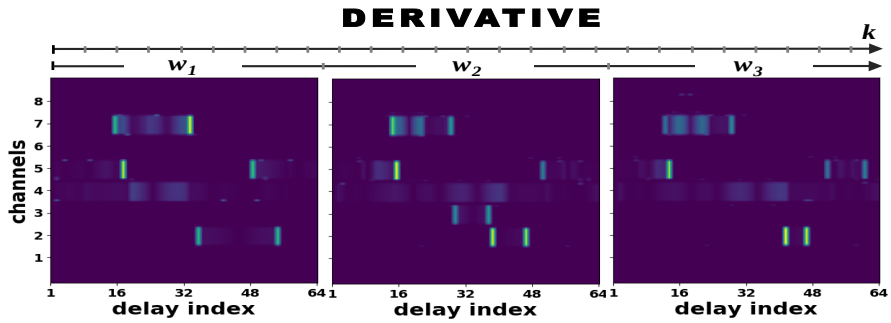
During this phase, both the G optimizer and the D optimizer are switched off to perform testing on new data samples. In this way, the parameters of the two networks are not updated, and the C-GAN can be used as an abnormality detector.
***Testing data*:** differently from the data with normality behaviour discussed in Sec. 7.2, an abnormality pattern consists of:

- Modality 1: a fixed (admissible) signal is occupying *ch-4* and a moving signal sequentially jumps between *ch-5*, *ch-7*, *ch-2*, *ch-5*

as shown in Fig. 7.3(a). The malicious signal follows a different behaviour or trajectory in the spectrum which has not been observed during training. Consequently, the behaviour or strategy of the abnormal signal is not compatible with the pattern



(a) Amplitude: Modality 1



(b) Derivative: Modality 1

FIGURE 7.3: Independent data samples from bi-dimensional localized dynamic amplitude. Pattern used as abnormal data during testing phase ($w = 1, 2, 3$; 64 shifts in the ST; and $k = 1, \cdots, 192$) and corresponding to modality 1

that produced the model. The corresponding absolute values of the derivatives are shown in the images of Fig. 7.3(b). Moreover, additional modalities (from 2 to 4), corresponding to abnormal patterns where the signal is jumping among the available channels in different ways, have also been considered for validation purpose as follows:

- Modality 2: a fixed signal is occupying *ch-4* and a moving signal jumps between *ch-8*, *ch-2*, *ch-8*, *ch-2*.
- Modality 3: a fixed signal is occupying *ch-4* and a moving signal jumps between *ch-5*, *ch-7*, *ch-6*, *ch-5*.
- Modality 4: a fixed signal is occupying *ch-4* and a moving signal jumps between *ch-7*, *ch-5*, *ch-2*, *ch-1*.

and shown in Fig. 7.4 where only one sample is displayed (w_1) for each modality.

In the following, results from abnormality detection related to the two different scenarios, described in Sec. 7.2, are presented where the abnormality indicator is described in Sec. 5.6.

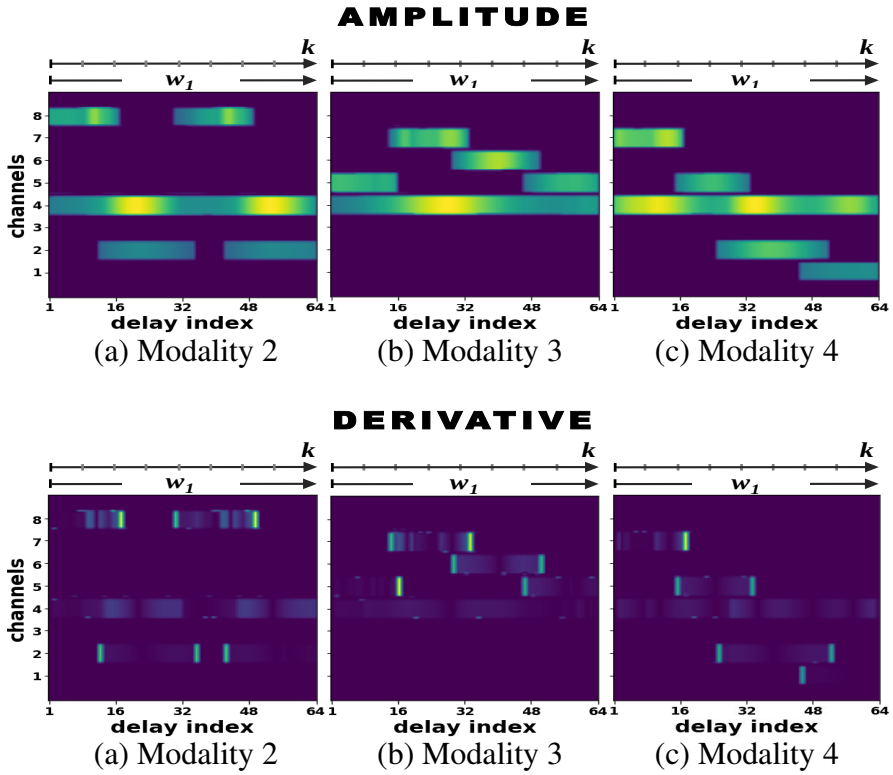


FIGURE 7.4: Three patterns used as abnormal data during testing phase and corresponding to modalities 2, 3, and 4, respectively, where only one sample is shown (w_1) for each modality

- **Scenario I** (matrix $\mathbf{X}_w$ used as image): during the online phase, $T = 1200$ image samples are tested. Each image is a matrix consisting of amplitudes in the time (rows) and frequency (columns) domain. An abnormality is detected whenever the signal is above the threshold, as shown in Fig. 7.5 (the squared regions) for each of the four modalities of the abnormal pattern. This is a promising result that confirms that the C-GAN could effectively be used as abnormality detector in a dynamic wireless scenario where a wideband spectrum consists of multiple signals and a specific feature (like amplitude in this work) is represented as an image, namely a two-dimensional representation.

- **Scenario II** (generalized state vector $\mathbf{x}_k$): in this scenario $T = 25280$ vector samples from the matrix representation of the amplitude (as feature) are tested during online phase. Again, an abnormality is detected whenever the signal is above the threshold. This can be seen in Fig. 7.6 for each of the four modalities of the abnormal pattern. Consequently, even when the C-GAN is given a generalized state vector as input, the model is capable of detecting abnormal patterns in which malicious behaviour produces deviations of predictions from observations. This would be a novel result because a generalized state vector is applied to a C-GAN model.

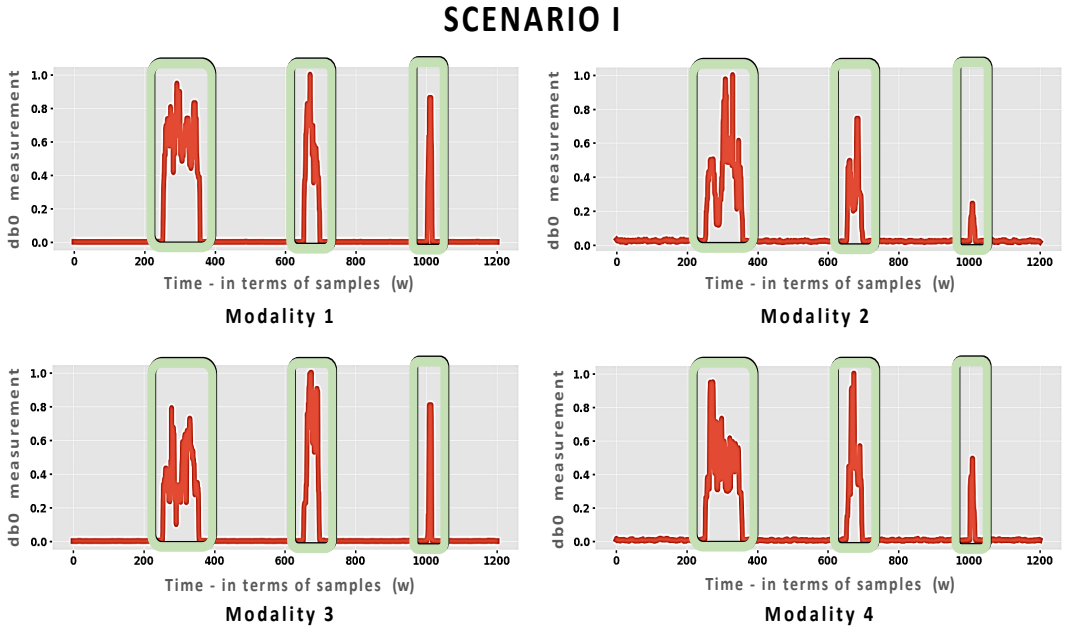


FIGURE 7.5: Normalized abnormality indicator (C-GAN model) at testing phase for modalities 1-4 with abnormal pattern detection when the input data consists of bi-dimensional localized dynamic amplitude as images ($w = 1, 2, \cdots, 1200$)

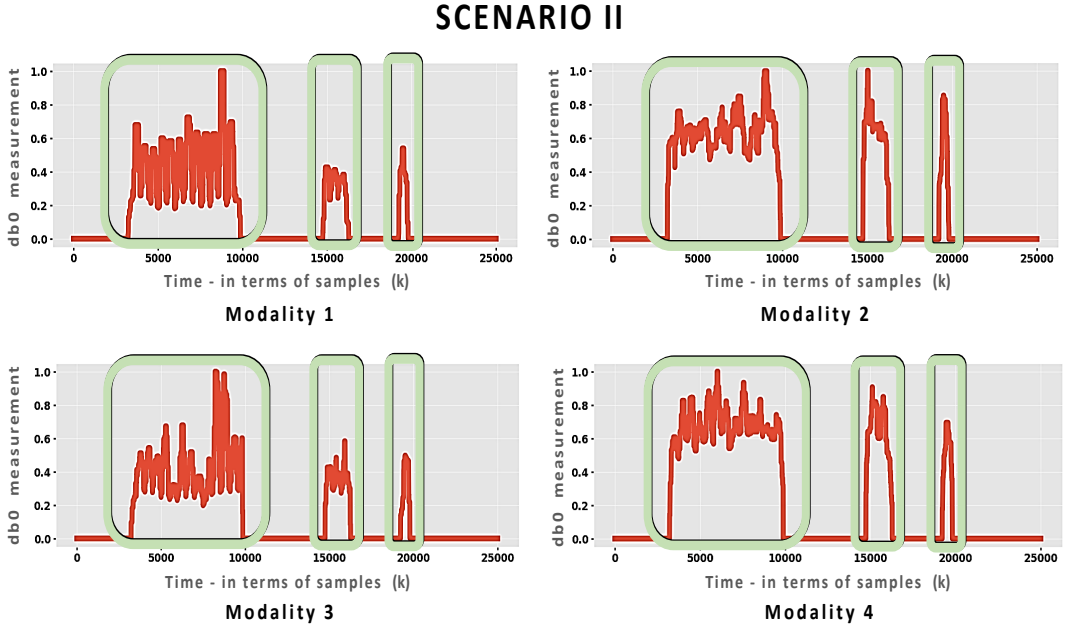
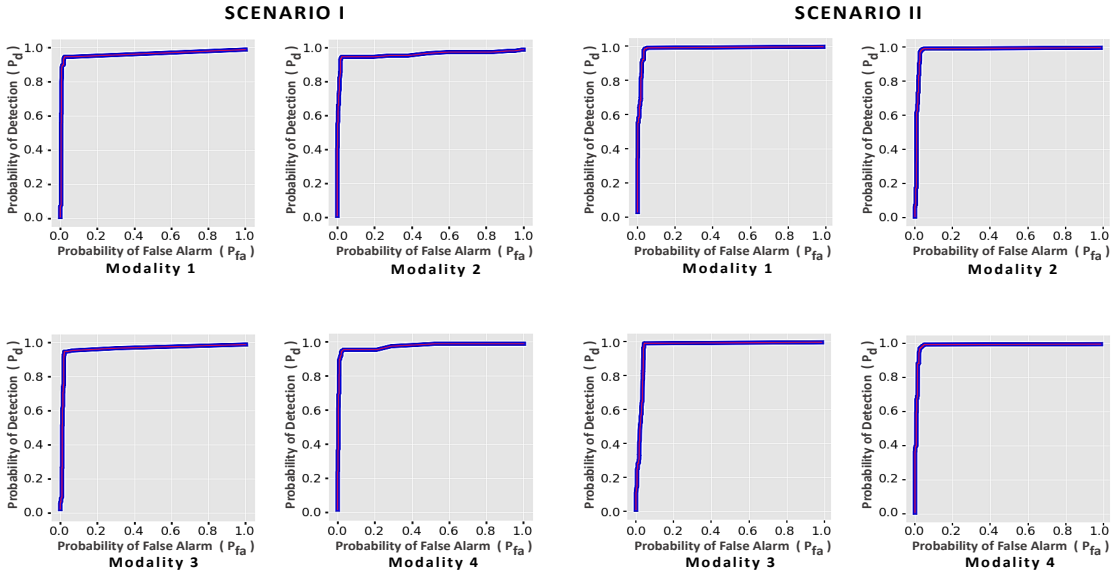


FIGURE 7.6: Normalized abnormality indicator (C-GAN model) at testing phase for modalities 1-4 with abnormal pattern detection when the input data is a generalized state vector with amplitude feature and its derivative ($k = 1, 2, \cdots, 25280$)

These results can be analyzed by considering that groups of samples in the testing data could exhibit different types of features (abnormal situation) from the ones observed during the training of the generative network with normality data [7]. In other words, since no neuron in the latent space was trained to detect these features, abnormality data cannot activate any neuron in the neural networks and the consequent deviation of prediction from observation produces high values of the abnormality measurements as shown in Figs. 7.5 and 7.6. Additionally, the ROC curves in Figs. 7.7(a) and 7.7(b) for the two scenarios confirm that the C-GAN can provide high detection probability with low P_{fa}. In addition, the P_d can be optimized through a sensible choice of the threshold in the binary testing.



(a) Scenario I (b) Scenario II

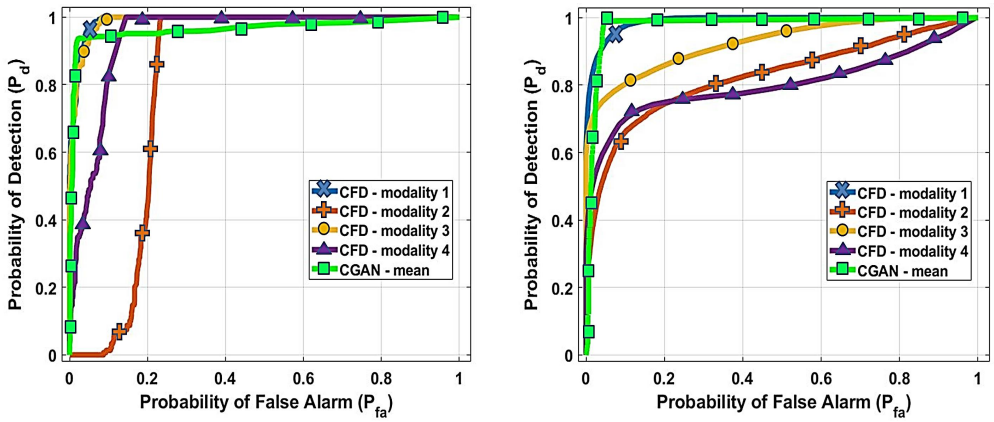
FIGURE 7.7: ROC curves corresponding to modalities 1-4 of the abnormal signal for the two scenarios (C-GAN model)

Before concluding, it should be summarized that the two performance metrics, abnormality indicators and ROC curves described in Sec. 6.3.4, can help to detect abnormalities and evaluate the performance of the model. According to ROC curves, the SA module can decide whether to incrementally learn new generative models whenever it performs poorly (in case of low P_d and high P_{fa}) or to act accordingly to mitigate malicious attacks.

7.4 Comparison with other methods

7.4.1 Comparison with conventional methods

Among the others, two conventional approaches that can be employed to detect abnormality signals inside the radio spectrum in CR are Energy Detector (ED) and Cyclostationary Feature Detection (CFD). ED is one of the simplest and most popular sensing techniques which compares the energy of the received signal with a certain threshold to decide whether the signal is present or absent. However, it suffers from the noise uncertainty [12]. In the CFD, the α-profile is extracted from the Spectral Correlation Function (SCF) of the observed signals to be used as a feature, as discussed in Sec. 3.3. This method is computationally complex and requires knowledge of prior information of the signal as the cyclic frequency is not always possible in real applications [6]. In this section, the performance of the C-GAN is compared with the conventional CFD and shown in Fig. 7.8. Specifically, Fig. 7.8 contains the ROC curves corresponding to the 4 analyzed modalities, as discussed in Sec. 7.3, for the two methods (C-GAN vs. CFD) and the two scenarios. 'C-GAN mean' represents the mean ROC curve over the 4 modalities from Figs. 7.7(a) and 7.7(b) (the mean is plotted since the ROC curves related to the 4 modalities are somehow similar). It is obvious from Fig. 7.7 and Fig. 7.8 how stable the performance of C-GAN and unstable that of CFD are. The corresponding Accuracy (ACC) values are listed in Tab. 7.1. Experiments show that the proposed framework outperforms the conventional detection method due to its twofold ability to predict the signal states (not only to detect them as in the conventional approach) and, consequently, to detect abnormalities resulting in a reduced amount of appearances of false alarms.



(a) Scenario I (C-GAN) (b) Scenario II (C-GAN)

FIGURE 7.8: C-GAN vs. CFD

	*db*0 Scenario I	*CFD* Scenario I	*db*0 Scenario II	*CFD* Scenario II
ACC (modality 1)	**0.9746**	0.9624	**0.9938**	0.9434
ACC (modality 2)	**0.9754**	0.8824	**0.9862**	0.8262
ACC (modality 3)	**0.9787**	0.9639	**0.9951**	0.8950
ACC (modality 4)	**0.9762**	0.9059	**0.9960**	0.8424

TABLE 7.1: Comparison in terms of accuracy measures between C-GAN and CFD

7.4.2 Comparison with AI-based methods

As claimed in Sec. 6.1, the main differences with respect to the cited AI-based techniques are that: *i)* the framework in this thesis is based on learning generative dynamic models by using the generalized state vector to incorporate both the state and its dynamics (namely the corresponding derivatives), which facilitates the prediction of the spectrum's future state and allows to detect any abnormality. Forming the generalized state vector is more efficient and less complicated by relying on raw I/Q data compared to other methods that deal with more complex features. *ii)* the proposed framework covers an application that can be implemented at the base station-side which deals with multi wideband signals. *iii)* As demonstrated with the considered OFDM case, the approach can be applied to generic wideband modulations providing in this way a flexible security tool for enabling CR-receivers with AI capabilities.

7.5 Abnormality detection through AC-GAN and VAE

Only Scenario II (which contributes with the most interesting novelty to this work) is investigated and a comparison with C-GAN is also performed. The ***training*** data consists of $T = 59520$ samples (indexed by k denoting the time domain in the ST representation), i.e. T observations of the generalized state vector $\mathbf{x}$ consisting of 256 elements (128 amplitude and 128 derivative relative to the frequency domain in the ST) for C-GAN, while $T = 164480$ samples (i.e. T observations of $\mathbf{x}$ with 256 elements) for AC-GAN and VAE. Again, by providing the normality data, the generative models are learnt in an unsupervised way. In the ***testing*** phase, $L = 25280$ samples (indexed by k denoting the time domain in the ST representation), i.e. L observations of the generalized state vector $\mathbf{x}$ with 256 elements (128 amplitude and 128 derivative relative to the frequency domain in the ST) are tested and an anomaly measurement is obtained (Figs. 7.9-7.10-7.11) for each of the three models and three analysed modalities (Modality 1-Modality 4-Modality 3 in Figs. 7.3-7.4). Again, it can be seen that, when the deep model is given a generalized state vector as input, it is capable of detecting the abnormal patterns in which malicious behaviour produces

deviations of predictions from observations. This would be a novel approach based on a generalized state vector applied to a deep generative model.

Additionally, the ROC curves are shown in Fig. 7.12 that confirm that each of the deep model can provide high detection probability with low P_{fa}. ROC curves can help to evaluate the performance of the three models. Indeed, Area Under ROC Curve (AUC) and Accuracy (ACC) values are extracted and listed in Tab. 7.2 where

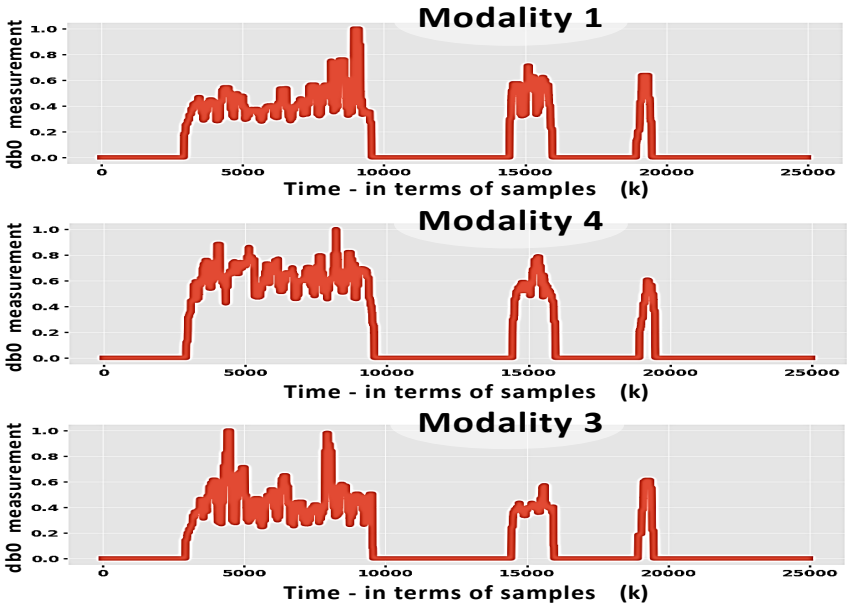


FIGURE 7.9: Anomaly indicator (**C-GAN model**)

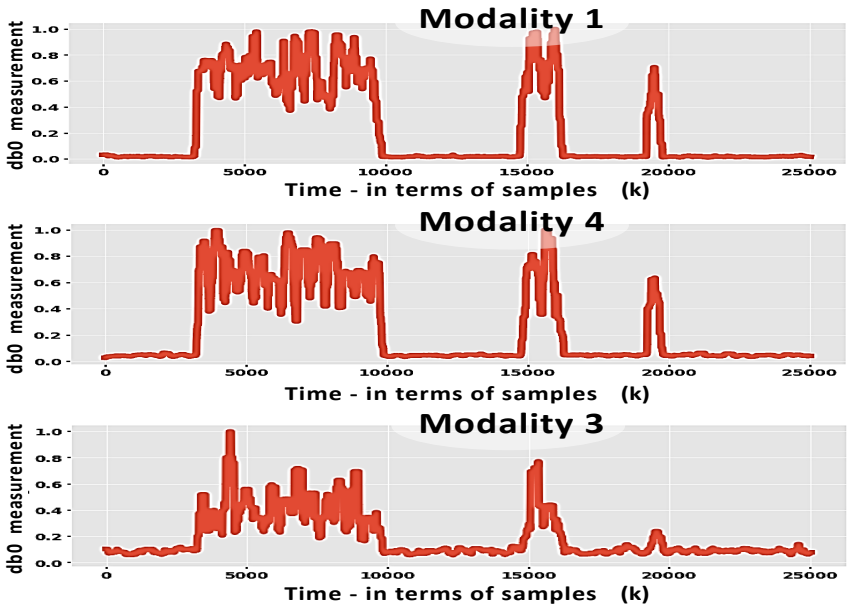


FIGURE 7.10: Anomaly indicator (**AC-GAN model**)

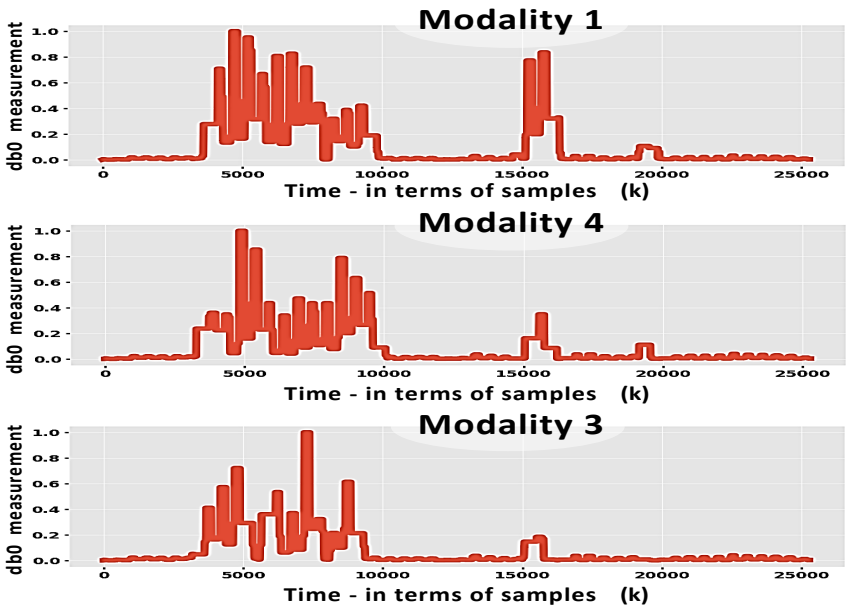
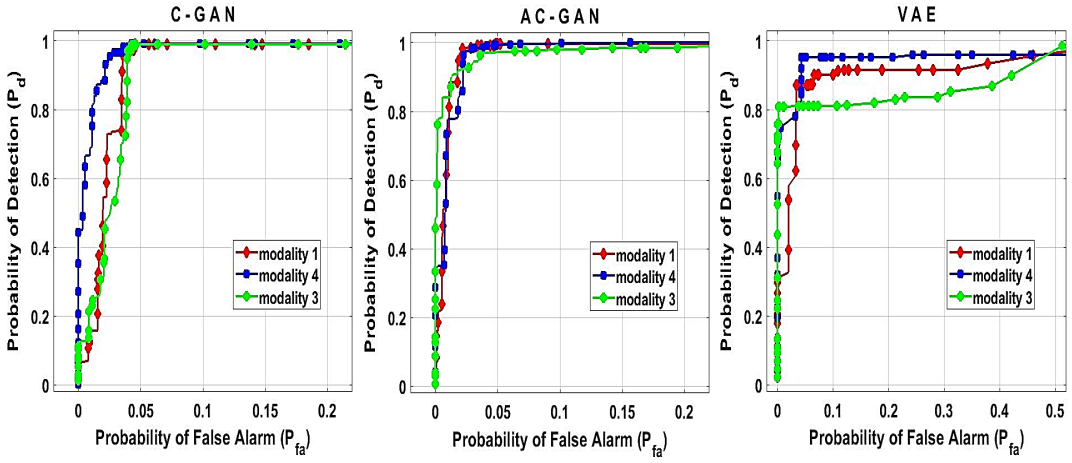


FIGURE 7.11: Anomaly indicator (**VAE model**)



(a) C-GAN model (b) AC-GAN model (c) VAE model

FIGURE 7.12: ROC curves

		AUC	ACC
C-GAN	**modality 1**	0.9566	0.9657
	modality 4	0.9737	0.9696
	modality 3	0.9545	0.9668
AC-GAN	**modality 1**	**0.9741**	**0.9804**
	modality 4	**0.9751**	**0.9757**
	modality 3	**0.9742**	**0.9660**
VAE	**modality 1**	0.9365	0.9356
	modality 4	0.9577	0.9551
	modality 3	0.9232	0.9382

TABLE 7.2: AUC and ACC values for the 3 Deep Learning Models

the AC-GAN seems to provide better performance than C-GAN and VAE models. From another point of view, when GAN-based models are compared to VAE, it can be noticed that: in the first case, since the generator is trained to learn a mapping between a random noise vector, $\mathbf{z}$ in Fig. 5.7, and the generated data (by learning hidden, complex structure in the real data $\mathbf{x}$), then G is able to capture the dynamics in the real data. In the second case, a VAE model returns the posterior probability that an observation belongs to a specific cluster by learning the latent vector, $\mathbf{z}$ in Fig. 5.8. In this way, observations $\mathbf{x}$ from different clusters will correspond to different $\mathbf{z}$ vectors and the dynamics of $\mathbf{x}$ is captured according to the way and the time instants the vector $\mathbf{z}$ changes. In effect, learning from dynamic data as in the first case should provide better performance as confirmed by the results. Alternatively, an advantage of the VAE, with respect to GAN, is the possibility to exploit the encoder's output latent variables ($\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$) that represent probabilistic distributions. Indeed, such variables can be clustered to learn temporal dependencies among them and draw a probabilistic graphical representation; for example, by using a Self Organizing Maps (SOM) method [4, 5] or a Growing Neural Gas (GNG) network [2]. The latent variables can also be used to reduce the complexity due to high dimensionality data in wideband RF spectrum.

Finally, Tab. 7.3 gives an idea about the time required to train and test the models under investigation. Among the 3 analysed models, VAE required less computational time to perform both training and testing processes, since KL is faster than MSE and L^p methods.

Deep Learning Models	**Training time [mm:ss]**	**Testing time [mm:ss]**
C-GAN	15:16	01:36
AC-GAN	30:42	03:16
VAE	**15:09**	**01:00**

TABLE 7.3: Computational times for the 3 Deep Learning Models

7.6 Conclusion and future work

The proposed work has investigated a framework which is foreseen to support the PHY-layer security in CR by introducing a basic Self-Awareness module; this module includes the capability of learning dynamic generative models and, consequently, detecting any abnormal signals inside the wireless spectrum. The potential of the proposed approach based on learning a C-GAN, AC-GAN, or VAE model lies in a fact that it can be incorporated in a CR system where the cognitive capability is close to the receiving antenna (before the demodulation process) and high-dimensional data

is extracted from the ST representation of the observed wideband spectrum consisting of multiple dynamic signals whatever their specific modulation scheme. Such an input data to the generative model is organized in a generalized state vector, with high dimensionality, which incorporates the signal amplitude and the corresponding derivative from the ST representation of the dynamic spectrum.

Generative models are capable of both generating synthesized data and providing a distance metric to measure the deviation of the predicted data from the observed one. Validation is performed on a real mmWave dataset. Extensive experiments have been conducted and a comparison analysis is proposed between the three deep generative models; in particular, the obtained results reveal that the proposed method can effectively detect spectrum abnormalities such as malicious or jammer attacks after learning the corresponding generative model when a generalized state vector represents the data from the dynamic spectrum. Indeed, in all the tested modalities, abnormality measurements have showed excellent performance for the three methods in the proposed framework, particularly the AC-GAN. ROC curves and the corresponding ACC/AUC values confirmed that the probability of detection is high with a low false alarm probability. From computational time analysis, the VAE resulted in operating faster than the other two networks.

Unless an increase in the complexity of the proposed deep model-based abnormality detection approach with the mmWave frequency range, as discussed in Sec. 6.3, the achieved results can be extended to other mmWave bands, different from the 28 GHz band, even where the spectrum availability is much greater than the 800 MHz in the analysed band. Indeed, the features in the generalized state vector would not depend much on the portion of the spectrum in the mmWave (mentioned in Sec. 2.2.3) where they are extracted.

However, the techniques and results in this work lead to the conclusion that further SA functionalities, like incremental learning and abnormality mitigation, can be achieved; these issues will be investigated in future work. In addition, the proposed approach will be employed to characterize and classify anomalous signals.

Bibliography

[1] C. Chen, O. Kedem, C. R. C. M. da Silva, and C. Cordeiro. Millimeter-Wave Fixed Wireless Access Using IEEE 802.11ay. *IEEE Communications Magazine*, pages 1–7, 2019.

[2] B. Fritzke. A growing neural gas network learns topologies. In *Advances in Neural Information Processing Systems 7*, pages 625–632. MIT Press, 1995.

[3] Q. Huang, X. Xie, H. Tang, T. Hong, M. Kadoch, K. K. Nguyen, and M. Cheriet. Machine-Learning-Based Cognitive Spectrum Assignment for 5G URLLC Applications. *IEEE Network*, 33(4):30–35, July 2019.

[4] T. Kohonen. Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43(1):59–69, 1982.

[5] T. Kohonen. The self-organizing map. *Proceedings of the IEEE*, 78(9):1464–1480, Sep. 1990.

[6] A. Martian, B. T. Sandu, O. Fratu, I. Marghescu, and R. Craciunescu. Spectrum sensing based on spectral correlation for cognitive radio systems. In *2014 4th International Conference on Wireless Communications, Vehicular Technology, Information Theory and Aerospace Electronic Systems (VITAE)*, pages 1–4, May 2014.

[7] A. Nguyen, J. Yosinski, Y. Bengio, A. Dosovitskiy, and J. Clune. Plug & play generative networks: Conditional iterative generation of images in latent space. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[8] Y. Song, W. Yang, X. Yang, Z. Xiang, and B. Wang. Physical Layer Security in Cognitive Millimeter Wave Networks. *IEEE Access*, 7:109162–109180, 2019.

[9] S. Vuppala, Y. J. Tolossa, G. Kaddoum, and G. Abreu. On the Physical Layer Security Analysis of Hybrid Millimeter Wave Networks. *IEEE Transactions on Communications*, 66(3):1139–1152, March 2018.

[10] K. Xiao, W. Li, M. Kadoch, and C. Li. On the Secrecy Capacity of 5G MmWave Small Cell Networks. *IEEE Wireless Communications*, 25(4):47–51, AUGUST 2018.

[11] H. Zhao, J. Zhang, L. Yang, G. Pan, and M. Alouini. Secure mmWave Communications in Cognitive Radio Networks. *IEEE Wireless Communications Letters*, 8(4):1171–1174, Aug 2019.

[12] Shilian Zheng, Shichuan Chen, Peihan Qi, Huaji Zhou, and Xiaoniu Yang. Spectrum Sensing Based on Deep Learning Classification for Cognitive Radios. *arXiv e-prints*, page arXiv:1909.06020, Sep 2019.

8 Conclusion of this thesis and future work

A practical application of spectrum abnormality detection at the PHY-layer in Cognitive Radio, along with the theoretical framework, is provided in this thesis according to the recent concept of autonomic computing systems defined in the scientific community as *"systems with greater levels of autonomy and the ability to manage themselves"*. In particular, the major objective of this thesis is to investigate a framework that can enable a certain degree of self-awareness as cognitive capability for cognitive radio devices and networks to support the PHY-layer security through a brain-inspired approach. Indeed, the proposed approach follows an Artificial Intelligence-based scheme where learning deep generative models, to represent the dynamic spectrum and signals, can effectively provide implementation of data-driven SA functionalities.

Observation of the Radio Environment through the PHY-layer which acts as Radio-to-Device interface enabling Spectrum Sensing and Feature Extraction, *Learning* dynamic models from the data by using Machine Learning and AI techniques, and *Reasoning* for data-driven spectrum abnormality detection compose the two functionalities investigated in this thesis as part of the basic data-driven SA module described in the Introduction. Specifically, C-GAN, AC-GAN, and VAE have been proposed as deep generative architectures for abnormality detection from features, organized in a generalized state vector, related to ST representation of the observed dynamic and multi-signal spectrum. To this end, a NI mmWave testebed with SDR components has been used to collect a real, dynamic, and wideband spectrum in the 28 GHz band.

Fundamentally, by looking into the future, the milestones towards intelligent radio devices can be listed as: *Self-Aware - Adaptive - Cognitive - Intelligent*. Each capability plays a key role in the development of its succeeding capability.

In this framework, pioneering work is conducted by Karl J. Friston who combines recent formulations of neuronal processing to provide an account of cognitive dynamics from basic principles. He has formulated flows in random dynamical systems in *generalized coordinates of motion* and exploited (filtering) procedures in statistics and control theory to demonstrate (neuronal) filtering in the brain. His work is inspired by the peculiar resistance of biological systems to the dispersive effects of external

fluctuations. This adaptive behaviour can be explained in terms of minimizing an upper (*free energy*) bound on the surprise (negative log-likelihood) of sensory samples.

"Biological systems act on the world to place an upper bound on the dispersion of their sensed states, while using those sensations to infer external states of the world. The resulting active inference is closely related to formulations in embodied cognition and artificial intelligence ... any system that exists will appear to minimize free energy and therefore engage in active inference."

According to Friston's work, any dynamics can be expressed in terms of a *Lagrangian or probabilistic model* of flow in generalized coordinates of motion.

However, after *learning of Generative Models* and *detection of Abnormal Behaviours* from spectrum observations, the gathered information will enable the third functionality of the basic data-driven SA module. In particular, in future work regarding **Stage 3**, the detected abnormalities can be used by the control system to enable *Abnormality Mitigation* through actions on the sensory states to reduce the surprise level, or the SA module itself uses the detected abnormalities to perform *Incremental Learning* to learn new models that represent dynamic situations that are different from the normal experience. Abnormality mitigation will minimize deviations between sensory input and the generative model, while incremental learning will increase previous experience that generated the learned normality model. Both of the strategies aim at minimizing the free energy. Information from ROC curves can also be used by the cognitive device and determines the abnormality detection accuracy of the model that is in charge to represent the dynamics of the spectrum.

Awareness of the spectrum can also be enhanced by exploiting information about the abnormal signals. Indeed, in addition to Automatic Modulation Recognition techniques, which have been attracting researchers in Cognitive Radio to differentiate signals with different modulation schemes, a general *signal re-identification* framework can be implemented through deep learning techniques. In this case, it is possible to recognize the malicious signals in the wideband spectrum according to their features and re-identify them after any changes in the parameters such as frequency or modulation.

Furthermore, extensions of VAE can be considered as future work for **Stage 2**. Specifically, to model temporal dependencies of the data, variational recurrent neural networks (VRNN), based on merging variational methods with deep learning techniques, and Long Short-Term Memory (LSTM), which is a recurrent neural network designed to handle long- and short-term data dependencies, can be investigated.

Concerning **Stage 1**, *compressed sampling* (CS) could be integrated in the spectrum sensing phase. Indeed, due to the wideband spectrum which should be observed in CR applications, very high sampling rate is required which greatly increases the computational complexity of the spectrum sensing algorithm. This motivates the development of sub-Nyquist techniques for reducing the operational sampling rate while retaining the spectral information. In particular, CS theory states that certain signals can be recovered from much fewer samples or measurements than in traditional sampling methods. CS basically combines the following key concepts: (i) sparse representation, in a specific domain, of the signal to be sampled with a choice of a linear basis for the class of the desired signal, and (ii) incoherent measurements of the considered signal to extract the maximum information using the minimum number of measurements. Specifically, in CR applications it is assumed that the spectrum utilization is low and hence the sensed wideband spectrum is sparse in the frequency domain. Motivation for this approach is exploiting CS in the ST transform to reduce the complexity of spectrum sensing, based on time-frequency analysis, and feature extraction. After sub-Nyquist sampling, the wideband signal can be recovered from these samples by using one of several possible recovery algorithms such as the conventional (convex) l_1-norm optimization method. However, the main problem is the computational complexity required by this recovery algorithm which appears to be almost prohibitive for real-time applications, especially when time-frequency analysis is included in the compressed sensing framework. Alternative techniques have been investigated in the literature for both compressed sampling and signal reconstruction.

To conclude, let us now focus on the **Radio-to-Device interface**. In this thesis, static CR devices have been considered whose position does not change over the time. A different scenario consists of mobile CR devices with non-zero velocity where *fading* can affect the communication channels between transmitters and receivers. Moreover, *multipath* in such scenario can deteriorate the signal-to-noise ratio at the receivers. These issues challenge the spectrum abnormality detection and should be taken into account in systems like IoT, V2X, or in a LTE cellular network where eNodeBs could be exploited to send control commands to Unmanned Aerial Vehicle (UAV) devices about instantaneous position and direction of their trajectory. Such communications can also be altered by malicious attacks.

In a learning formulation, there are three components (let us consider the notation adopted in VAE): the data $\mathbf{x}$, the likelihood of the data $p_{\boldsymbol{\theta}}(\mathbf{x})$, and our predicted model $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$ which should be an approximation of the likelihood function. The distribution $p_{\boldsymbol{\theta}}(\mathbf{x})$ represents the way in which the data is observed and can be a Gaussian distribution in AWGN channels. Each of the three components could produce an abnormality detection. Indeed, in the scenario with moving devices, the abnormality

could be the effect of fading and multipath on the observed data even if the true data is normal. In other words, free energy minimization is not reached and, then, the data is observed as it were abnormal (this is related to perception through the sensory states). More formally, there are there possible *sources of abnormality*: the spectrum ($\mathbf{x}$), the predicted model ($q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$), and the channel condition ($p_{\boldsymbol{\theta}}(\mathbf{x})$). In this thesis spectrum abnormality is investigated. In the case of abnormality detection due to the channel condition, the likelihood model $p_{\boldsymbol{\theta}}(\mathbf{x})$ should be changed to take into account fading and multipath. This is a kind of *channel estimation*, which would be performed just after down-conversion and before demodulation of the received signal. It is still an open issue and application of machine learning to solve this problem has not been investigated yet.