DISTRIBUTED REINFORCEMENT AND DEEP LEARNING FOR SPECTRUM MANAGEMENT IN COGNITIVE RADIO NETWORKS

By Dejan Dašić

WORD COUNT 36189 TIME SUBMITTED 23-DEC-2024 05:15PM
PAPER ID 113690161

SINGIDUNUM UNIVERSITY

BELGRADE

DEPARTMENT FOR POSTGRADUATE STUDIES

DOCTORAL THESSIS

DISTRIBUTED REINFORCEMENT AND DEEP LEARNING FOR SPECTRUM MANAGEMENT IN COGNITIVE RADIO NETWORKS

MENTOR:

Prof. Miloš Stanković, PhD

STUDENT: Dejan Dašić

STUDENT NUMBER: 466072/2018

BELGRADE, 2024.

INFORMATION ABOUT THE MENTOR AND MEMBERS OF THE COMMISSION

Mentor:

Prof. Miloš Stanković PhD, mentor, Full Professor,
Singidunum University, Belgrade

Commitee Members:

Prof. Miljan Vučetić, PhD, co-mentor, Principal research fellow, Vlatacom Institute of High Technologies, Belgrade; Full professor, Singidunum University, Belgrade

Prof. Nebojša Bačanin-Džakula PhD, Full professor, Singidunum University, Belgrade

Prof. Mladen Veinović, PhD, Full professor, Singidunum University, Belgrade

Prof. Nemanja Ilić, PhD, Senior research associate, Vlatacom Institute of High Technologies, Belgrade; Associate Professor, Faculty of Computing, Belgrade

Dissertation defense date:

_____

ACKNOWLEDGMENT

I would like to thank my mentor, Prof. Miloš Stanković, PhD, co-mentor Prof. Miljan Vučetić, PhD, as well as Prof. Nemanja [illegible]ć, PhD, for their continuous support, valuable advice, and active participation during the preparation of this doctoral dissertation.

I owe a great debt of gratitude to the Vlatacom Institute, which provided me with the opportunity, time, and resources to conduct research, as well as invaluable support during the preparation of this dissertation. I would also like to thank all the members of the team that I worked with on the research, development and implementation of the projects of the Vlatacom Institute, from which this dissertation emerged.

ABSTRACT

Cognitive radio networks offer a promising solution to address the growing demand for wireless spectrum by enabling dynamic spectrum access. However, effective spectrum sensing and channel selection in such networks remain challenging due to the highly dynamic and uncertain nature of the wireless environment. This dissertation focuses on the development of a novel algorithm for distributed spectrum sensing and communication channel selection, based on consensus mechanisms within cognitive radio networks. The proposed approach leverages the collective intelligence of multiple agents using reinforcement learning, allowing them to learn optimal strategies for spectrum sensing while achieving consensus on channel selection in a decentralized manner.

The proposed consensus strategy enables decentralized and distributed cooperation among agents through a directed, time-varying communication network with low bandwidth. The motivation for this approach arises from typical scenarios in cognitive radio networks, where such decentralized setups and distributed operations are of critical importance. The proposed framework equips all agents with actionable information about network states, even in unknown environments and operational conditions. As a result, the participating agents in the network collectively achieve an optimal strategy for spectrum sensing and channel selection, even though individual agents may lack the capability to do so on their own. The algorithm is designed to be scalable and robust to agent or link failures.

The dissertation examines key characteristics of the algorithm, including its denoising effects, capacity for coordinating actions, and improvements in convergence rates achieved through consensus mechanisms. The findings contribute to the development of more robust and scalable solutions for cognitive radio networks, with applications in next-generation wireless communication and Internet of Things systems. Additionally, it explores the application of deep learning techniques to address challenges associated with modern spectrum management in new generation of radio networks, for providing the requested level of services.

REZIME

Kognitivne radio mreže nude obećavajuće rešenje za rešavanje rastuće potražnje za bežičnim spektrom omogućavanjem dinamičkog pristupa spektru. Međutim, efikasno upravljanje spektrom i odabir kanala u takvim mrežama ostaju izazovan problem zbog visoke dinamičnosti i neizvesnosti samog okruženja. Ova doktorska disertacija fokusira se na razvoj novog algoritma za distribuirano praćenje zauzetosti spektra i izbor komunikacionih kanala, zasnovanog na mehanizmima konsenzusa unutar kognitivnih radio mreža. Predloženi pristup koristi kolektivnu inteligenciju više agenata koji primenjuju učenje potkrepljenjem, omogućavajući im da nauče optimalne strategije za upravljanje spektrom, dok postižu konsenzus u odabiru kanala na decentralizovan način.

Predložena strategija konsenzusa, implementirana na usmerenoj, vremenski promenljivoj nisko propusnoj komunikacionoj mreži, olakšava decentralizovanu i distribuiranu saradnju među agentima. Motivacija za ovaj pristup proizilazi iz tipičnih scenarija u kognitivnim radio mrežama, gde su takva decentralizovana postavka i distribuirani rad od kritične važnosti. Predloženi okvir oprema sve agente, , sa informacijama o stanju mreže koje se mogu primeniti, čak i u nepoznatim okruženjima i operativnim uslovima. Shodno tome, agenti koji učestvuju u mreži kolektivno postižu optimalnu strategiju za upravljanje zauzetosti spektra i izbor kanala, iako pojedinačni agenti možda nemaju sposobnost da to urade sami. Algoritam je dizajniran da bude skalabilan i otporan na greške agenta ili veze između njih.

U disertaciji su ispitane ključne karakteristike algoritma, uključujući efekat smanjenja šuma, kapacitet za koordinaciju akcija i poboljšanje u brzini konvergencije postignutim putem mehanizama konsenzusa. Rezultati ove disertacije mogu da doprinosu razvoju robusnijih i skalabilnijih rešenja za kognitivne radio mreže, sa primenama u mrežama bežičnih komunikacija nove generacije i sistemima Internet stvari. Takođe, u disertaciji se razmatra primena tehnika dubokog učenja kako bi se rešavali izazovi povezani sa savremenim upravljanjem spektrom u radio mrežama nove generacija, radi obezbeđivanja traženog nivoa servisa.

TABLE OF CONTENTS

INFORMATION ABOUT THE MENTOR AND MEMBERS OF THE COMMISSION ... 2
ACKNOWLEDGMENT ... 3
ABSTRACT ... 4
REZIME ... 5
TABLE OF CONTENTS ... 6
LIST OF ABBREVIATIONS ... 8
LIST OF FIGURES ... 10
LIST OF TABLES ... 11
1. INTRODUCTION ... 12
1.1 Hypotheses, Objectives and Research Methods ... 14
1.2 Related Work ... 16
1.3 Scientific Contribution ... 18
2. COGNITIVE RADIO NETWORKS ... 22
2.1 CRN Introduction ... 22
2.2 Key Concepts in Cognitive Radio Networks ... 23
2.3 Cognitive Radio Network Architecture ... 25
2.4 Cognitive Radio Applications ... 25
2.5 Standardizations ... 26
3. REINFORCEMENT LEARNING ... 28
3.1 Machine Learning ... 28
3.2 Reinforcement Learning Concepts ... 28
3.2.1 Markov Decision Process ... 30
3.2.2 Temporal-Difference Learning ... 32
4. DISTRIBUTED REINFORCEMENT LEARNING ... 33
4.1 Motivation and Approaches to Distributed Reinforcement Learning ... 33
4.2 Distributed Temporal Difference Algorithm ... 34
4.2.1 Convergence analysis ... 36
5. NEURAL NETWORKS and DEEP LEARNING ... 42
5.1 Overview of Neural Networks ... 43
5.2 Deep Learning Basics ... 45
5.2.1 Reducing the Deep Learning Model's Complexity ... 48
5.3 Deep Learning Architectures for Mobile Networks ... 50
5.4 Deep Learning Applications in Mobile Networks ... 52
5.4.1 Mobile Data Analysis ... 53

5.4.2 Network-Level Data 53
5.4.3 Application-Level Data 54
5.4.4 Mobility Analysis 55
5.4.5 User Localization 55
5.4.6 Network Control 55
5.4.7 Network Security 55
5.4.8 Signal Processing 57
5.5 The Limitations of Application of Deep Learning 58
6. JOINT SPECTRUM SENSING AND CHANNEL SELECTION 60
6.1 Problem Description and System Model 61
6.1.1 Joint Spectrum Sensing and Channel Selection 61
6.1.2 MDP Formulation of the JSS model 62
6.2 Consensus-Based Distributed Joint Spectrum Sensing and Selection 64
6.2.1 Distributed Consensus-based Policy Evaluation 65
6.2.2 Distributed Consensus-based Q-Learning 67
6.2.3 Convergence Rate and Complexity Analysis 68
6.3 Simulations 69
7. SPECTRUM SENSING BY MODULATION CLASSIFICATION USING DEEP NEURAL NETWORKS 81
7.1 Challenges in Spectrum Management and Utilization 81
7.2 Strategic Role of Spectrum Sensing 82
7.3 Integration of Deep Learning into Spectrum Sensing 82
7.4 Application of Deep Learning Frameworks in Spectrum Sensing 83
7.5 Performance Comparison: Traditional vs. Deep Learning Approaches 84
7.6 Challenges and Research Perspectives 85
7.7 Deep Learning-Based Modulation Classification Use Case 86
8. CONCLUSION 89
LITERATURE 92

LIST OF ABBREVIATIONS

AE - Autoencoder
ANN - Artificial neural network
ARIMA - Autoregressive Integrated Moving Average
ASR - Automatic Speech Recognition
BAN - Body Area Networks
CDR - Call Data Register
CG - Coordinated graph
CM - Cognition module
CN - Cognitive networks
CNN - Convolutional Neural Networks
CR - Cognitive Radio
CRN - Cognitive radio networks
CR-VANET - Cognitive Radio-Vehicular Ad Hoc Networks
CS - Cloud Servers
CSM - Cooperative spectrum management
CSS - Cooperative spectrum sensing
DAG - Directed Acyclic Graphs
DBN - Deep Belief Network
DCS - Dynamic channel selection
DL - Deep Learning
DP - Dynamic Programming
DQN - Deep Q-learning
DRL - Deep reinforcement learning
DRQN - Deep recurrent Q-networks
DSA - Dynamic Spectrum Access
DySPAN - Dynamic Spectrum Access Networks
ER - Experience replay
FNN - Feedforward Neural Networks
FS - Fog Servers
GAN - Generative Adversarial Networks
GP - Gaussian Processes
IoT - Internet of Things
ISC - Interference State Control
JSS - Joint spectrum sensing
LSTM - Long Short-Term Memory
MAB - Multi-armed bandit
MADDPG - Multi-Agent Deep Deterministic Policy Gradient (
MARL - Multi-agent reinforcement learning
MDLA - Mobile Deep Learning Applications
MDP - Markov decision process
MIMO - Multiple-Input Multiple-Output
ML - Machine learning
MLP - Multilayer Perceptrons
MSE - Mean Squared Error
NN - Neural networks
NPU - Neural processing units
ODE - Ordinary Differential Equation
PER - Packet Error Rate

PU - Primary User
QoE - Quality of Experience
QoS - Quality of Service
RBM - Restricted Boltzmann Machine
RL - Reinforcement learning
RNN - Recurrent neural network
SCF - Spectral Correlation Function
SDR - Software Defined Radio
SGD - Stochastic Gradient Descent
SS - Spectrum sensing
SU - Secondary User
SVM - Support Vector Machines
TD - Temporal difference
UCB-H - Upper confidence bound
V2I - Vehicle-to-infrastructure
V2V - Vehicle-to-vehicle

LIST OF FIGURES

Figure 5-1 Overview of relationship between Deep Learning, Machine Learning and Artificial Intelligence 46
Figure 5-2 Overview of Cloud, Fog, Edge and Mobile computing 52
Figure 5-3 Deep Learning application subject areas in mobile networks 53
Figure 6-1 MDS formulation [43] 63
Figure 6-2 Algorithm 1 - Distributed consensus-based policy evaluation [43] 67
Figure 6-3 Algorithm 2 - Distributed consensus-based Q-learning [43] 69
Figure 6-4 Estimations of the value function by all the agents (without consensus) are shown as different lines for all the states, which are shown on different plots. Times while the channels are occupied by primary users are shaded in red. [43] 72
Figure 6-5 Estimations of the value function by all the agents with consensus scheme implemented are shown as different lines for all the states, which are shown on different plots. Times while the channels are occupied by primary users are shaded in red [43] 73
Figure 6-6 Statistics of errors in estimations of state-value function for different algorithms as a function of the number of iterations [43] 74
Figure 6-7 Estimations of the value function performed by all agents both without consensus (upper figure) and with consensus (lower figure), shown with different lines, for a state which can only be visited by three agents. Times while the channel in question is occupied by primary users are shaded in red. [43] 75
Figure 6-8 Estimations of action-value function for one channel by all the agents (without consensus) [43] 77
Figure 6-9 Estimations of action-value function for one channel by all the agents (with consensus) 78
Figure 6-10 Total count of transmission events within the network, both successful and failed, for different algorithms as a function of the number of iterations [43] 79
Figure 7-1 Different deep learning models for different SNRs [163] 87
Figure 7-2 Performance of CNN deep LSTM [163] 88

LIST OF TABLES

Table 1. Overview or related work 18
Table 2 Channel parameters [43] 70
Table 3 Target policy parameters [43] 70
Table 4 Behavior policies parameters [43] 70
Table 5 Contributions through participating channels within the network to the total number of successful and failed transmissions [43] 80

1. INTRODUCTION

A Cognitive Radio (CR) is a highly flexible wireless transceiver designed to autonomously adjust its communication parameters to meet the demands of the network environment and user Quality of Service (QoS) requirements ([1], [2]). It integrates dynamic spectrum access (DSA) at the hardware level with intelligent decision-making enabled by a cognition module (CM) at the software level. Using DSA, CRs actively monitor the network environment and adapt transmission parameters such as frequency, modulation, coding, or power based on the decisions made by the CM [3], [4].

One key use case for CRs is the dynamic reuse of vacant portions of licensed spectrum by secondary users (SUs) while ensuring no interference with primary users (PUs) who own the spectrum [1]. Channel sensing techniques play a crucial role in detecting these transmission opportunities in both frequency and time domains [5]. In general, wireless channels are subject to shadowing and multipath fading, leading to detection errors by individual SUs. These errors may waste spectrum opportunities or cause interference with PUs. This challenge can be mitigated through cooperative sensing among SUs (e.g. [6]), advanced spectrum sensing algorithms at the DSA level [5] and CM-level collaboration [7]. The critical role of advanced machine learning (ML) techniques, such as reinforcement learning and cooperative approaches, in ensuring the efficient operation of CR has recently sparked growing interest in their applications and the development of cognition modules (CM) ([2], [8] and references therein).

Reinforcement learning (RL) is widely recognized for addressing decision-making problems in uncertain environments, often modeled as Markov decision processes (MDPs) [9]. RL is based on the concept of rewards, where an agent's actions influence the environment, resulting in state transitions and associated rewards. The agent aims to maximize cumulative rewards by adapting its actions based on the observed state and reward received for an action taken or a state reached. Traditional RL methods rely on approximate and adaptive dynamic programming, focusing on efficient computation or approximation of state-value or action-value functions [9]. Handling large state spaces, which are often encountered in analyses of these problems, often requires parameterizing or approximating these functions in lower-dimensional spaces. In addition, it is of considerable importance for a participating agent, especially in distributed multi-agent settings, such as in cooperative spectrum management (CSM) in CRs, to be able to evaluate a policy that differs from the one used for learning, which is known as off-policy learning. Recently, a number of gradient-based temporal-difference (TD) learning algorithms with these capabilities have been developed [10]. Multi-agent reinforcement learning (MARL) has emerged as a highly active area of research, particularly over the last decade. One notable MARL framework involves agents interacting through an underlying Markov Decision Process (MDP), where the decisions of one agent influence the decisions of others. This interaction can be analyzed within the context of stochastic games, which may be cooperative or non-cooperative in nature (see, e.g., [11] and references therein). Another interesting setting involves agents operating independently on separate realizations of the same MDP. In this decentralized framework, agents do not interact through the underlying process [12], [13]. This setup eliminates the need for a centralized fusion node, enabling fully decentralized operation. The algorithms proposed in [12], [13] allow

agents to maintain unique behavior policies while collectively evaluating the response of the MDP a single target policy. Such methods enable the entire group of agents to compute the value function successfully, even when individual agents lack sufficient capacity to do so.

Due to the simplicity and effectiveness of this decentralized approach, the focus of the current research proposal is its adaptation for Cooperative Spectrum Management (CSM) in Cognitive Radio (CR) systems. The resulting scheme, combining these advantageous properties, has the potential to significantly advance CR technology. Moreover, it could contribute to broader applications in cognitive networks (CN) and next-generation radio systems.

Modern mobile networks generate vast amounts of heterogeneous data, necessitating advanced methods for classification, prediction, and decision-making. Compared to traditional machine learning (ML), deep learning (DL) models are better suited to handle the inherent complexity of mobile network data due to their ability to capture intricate patterns and relationships. Additionally, DL algorithms excel in leveraging large datasets, which act as natural regularizers to prevent overfitting [14]. While training on massive datasets may require more time, this challenge is mitigated by the scalability of DL training algorithms, typically variants of Stochastic Gradient Descent (SGD). These methods efficiently process data in small batches during each training step [15].

Data generated by mobile networks are often noisy, heterogeneous, and exhibit complex spatial-temporal patterns [16]. In contrast to traditional ML models, DL models inherently extract feature hierarchies from such data with internal correlations [14], making them superior for the purposes of further data analyses over traditional ML approaches, which often rely on expensive and labor-intensive manual feature engineering [17].

The increasing preference for implementation of DL models in next-generation mobile networks also stems from their scalability and ability to handle unlabeled or semi-labeled data [16]. Modern DL techniques enable unsupervised learning to extract meaningful patterns, significantly reducing annotation costs. The work presented in this thesis considers in part the applications of such algorithms in the present and future mobile radio systems.

The integration of DL requirements into mobile network design has spurred the development of new architectures to support emerging services. Various infrastructure models—such as cloud, fog, edge, and mobile computing resources—have been proposed to optimize the implementation of DL applications [18]. These architectures influence the scalability, latency, availability, and communication capabilities of DL-powered applications within 5G and 6G networks. By leveraging the cognitive capabilities of DL, future mobile networks can enhance their functionality across diverse environments, paving the way for innovative and efficient network solutions.

This dissertation presents a novel algorithm for distributed joint spectrum sensing and channel selection, leveraging consensus among secondary users in cognitive radio networks. The algorithm is integrated within a multi-agent reinforcement learning framework. The proposed consensus strategy operates over directed, time-varying

communication networks with low bandwidth and throughput, enabling decentralized and distributed cooperation among agents. This approach is motivated by typical cognitive radio network scenarios, where decentralized setups and distributed operations are crucial. The proposed framework equips agents with actionable insights into the network's state, even in unknown environments and application conditions. As a result, the secondary users collectively determine the optimal joint strategy for spectrum sensing and channel selection, a task that individual agents possibly cannot achieve independently. The proposed algorithm is scalable and robust against failure of agents or links between agents. The dissertation includes consideration of algorithm features, such as the denoising effect, the ability to organize coordinated actions, and the improvement of the convergence rate that occurs due to consensus. As part of the consideration of channel sensing of the spectrum of interest, the dissertation also considers the application of DL (Deep Learning) as part of the response to challenges in this area.

1.1 Hypotheses, Objectives and Research Methods

The general hypothesis that started the research in the dissertation is:

H1. The utilization of spectrum opportunities for information transmission in cognitive networks can be improved in distributed multi-agent systems.

Specific hypotheses arising from the general are:

H1.1 If enough information is known about the primary spectrum users, deep learning can be used to identify spectrum opportunities.

H1.2 The utilization of spectrum opportunities for information transfer in cognitive networks can be improved by introducing consensus within reinforcement learning in distributed multi-agent systems.

The individual hypotheses used in the dissertation are:

H1.1.1 If the mode of operation (modulation) of the primary user of the radio spectrum is known, deep learning can be used for precise identification of the activities of the primary users in cognitive radio networks.

H1.2.1 The introduction of consensus in the setting of distributed multi-agent reinforcement learning in the treatment of spectrum management in case of cognitive radio networks brings an improvement in the speed of information flow about available channels, that is, it favorably affects the achieved information flow of secondary users of the system.

H1.2.2 Multi-agent systems in cognitive networks can achieve performance close to centralized systems in both distributed and decentralized settings.

The main objectives of the research were as follows:

O1. Formulation of the spectrum management problem in the context of a multi-agent reinforcement learning setting.

O2. Deep learning techniques to be applied to identify opportunities in the radio spectrum.

O3. Existing general distributed multi-agent reinforcement learning methods will be modified and extended in such a way that they can be applied to the spectrum management problem.

O4. The algorithms will be directly applied to the problem of spectrum management in cognitive radio networks (CRN).
O5. The application of deep learning techniques to evaluate the activities of primary users will be verified by simulations.
O6. A new spectrum management scheme based on multi-agent learning with consensus support will be verified by simulations.

In the course of scientific research, various methods were employed to meet fundamental methodological principles, including objectivity, reliability, generality and systematicity.

In accordance with the selected subject matter, defined research objectives and set scientific hypotheses, in order to define scientific and expert conclusions and find possible solutions, a theoretical analysis was used which included the research of results from international scientific literature, i.e., the knowledge of scientists and other authors who researched the issues that this dissertation deals with as well. The dissertation presents scientific-theoretical knowledge, relevant literature and original proposals using a number of methods, namely: historical methods, methods of complex observation and content analysis, methods of analysis and synthesis, methods of proof, as well as experimental methods.

By applying the historical method, the results of the research of other authors who dealt with issues related to the topic of interest were obtained. The obtained data comes mainly from eminent scientific works and research from Italy, India and China.

The method of complex observation and content analysis was applied when processing the results taken from the research of other researchers. The results were used to define the required achievements of the newly proposed schemes of multi-agent systems and corresponding algorithms, as well as the success of the deep learning scheme in terms of classification capabilities.

By studying the achievements and settings applied in existing research, the possibility of improving the existing and applying the improved settings was observed, by using new algorithms for the behavior of secondary users in cognitive networks modified to include cooperation based on the principle of consensus. Also, knowing the characteristics of the primary users, it is possible to define the setting of deep learning of the agents that would enable the precise identification of the occupancy of the frequency resource by the primary users. In this way, the method of analysis and synthesis was applied.

In order to analyze the correctness and applicability of the proposed schemes, a theoretical treatment is presented using the method of proving the existence of the desired properties, which were confirmed for the case of the assumed real conditions of implementation of the settings. In order to verify the effectiveness and efficiency of the proposed settings, behavioral simulations were performed that confirmed the initial hypotheses of the dissertation, which represents the implementation of the experimental method.

1.2 Related Work

Over the past decade, reinforcement learning, particularly multi-agent reinforcement learning, has emerged as a viable framework for addressing spectrum management challenges in cognitive radio networks (CRNs). A variety of MARL algorithms have been developed, many of which operate without assuming explicit cooperation between agents. These include methods that leverage value function approximations [19][20] or multi-armed bandit (MAB) approaches [21], [22] within their learning frameworks.

Cooperative MARL schemes have also been explored, involving either multiple fusion centers [6], [23], [24] or some degree of centralization [25], [26], [27]. Several of these approaches incorporate deep learning techniques, such as deep Q-networks (DQNs) [27], which have also been employed in single-agent RL scenarios [28]. Other RL-based methods extend beyond channel selection tasks to predict the duration for which a channel will remain unoccupied [29] or propose innovative hybrid spectrum access models [30].

Additionally, decentralized cooperative MARL algorithms have been developed, employing strategies like comparing shared information between agents [31] or integrating evolutionary game theory with DQNs [32]. One of the rare approaches to spectrum sensing in CRNs using a dynamic consensus scheme was presented in [33]. However, this approach addresses a less general problem that does not incorporate an RL-based channel selection task.

In [34], the authors employ deep recurrent Q-networks (DRQN) to handle partial observability of participating agents for the purpose of cooperative spectrum access, as DRQN incorporates recurrent neural networks (RNNs) to handle partially observable problems. RNNs are capable of retaining historical data, enabling them to infer the environment's state from sequential, albeit incomplete, information.

The authors in [35] employ the Multi-Agent Deep Deterministic Policy Gradient (MADDPG) algorithm, which features centralized training and decentralized execution in an actor-critic architecture, with multiple actors being cooperating SUs, in order to reduce the synchronization and communication overhead caused by the sensing cooperation of SU.

Expanding on this work, in terms of cooperation in spectrum sensing, in [36] the same authors introduced an attention mechanism for the purpose of selecting reliable partners for cooperative sensing within the CRN network (Table 1).

Related to the issue of application of deep learning in mobile and wireless networking, the authors in [15] provide a thorough review of the intersections between these areas. In [16], the research presents deep models learning framework for analytics of mobile big data within large-scale mobile systems based on a distributed computing platform. The work in [18] presents a guide on fog computing and the computing paradigms relating to ML within this realm. The research presented in [37] surveys applications of ML in the domain of radio signal processing, emphasizing the importance of appropriate datasets for model learning, testing, and evaluation and introduces several such synthetic datasets. Investigation of the performance of radio modulation

classification using features learned directly by CNN from the domain of complex-valued temporal radio signals is presented in [38], while the work in [39] discusses data augmentation methods for increasing of the volume and the quality of training data. Application of deep neural networks (NN) for the purposes of radio modulation recognition in investigated in [40].

Authors	Treated problem	Processing in local	Scheme of cooperation
C.Wu et al. [19]	Channel and power level selection	Q-learning	Centralized
T. Jiang et al. [20]	Efficient exploration in spectrum sharing	Linear value function approximation	None
Y. Tian et al. [21]	Channel and power level selection	MAB with policy index (energy efficiency)	None
N. Modi et al. [22]	Channel selection	MAB with learning policy (quality metric reflecting interference)	None
B.F. Lo, I.F. Akyildiz [6]	Cooperation overhead	Q-learning	Cooperative sensing
I. Mustapha et al. [23]	Cooperative channel sensing	Q-learning	Cluster heads aggregation
W. Ning [24]	Cooperative spectrum sensing	Q-learning	Partner selection algorithm
A. Kaur, K. Kumar [25]	Imperfect Channel State Information-based spectrum management	Q-learning	Assisted with cloud computing
S-J. Jang et al. [26]	Dynamic band and channel selection	Q-learning	Centralized
O. Naparstek, K. Cohen [27]	Distributed dynamic spectrum access	Deep Q-learning with LSTM	Centralized
S.Wang et al. [28]	Dynamic multichannel access	Deep Q-learning with Experience Replay	None
V. Raj et al. [29]	Channel selection and availability prediction	MAB setting and Bayesian learning	None
Y. Lin et al. [30]	Dynamic spectrum access, power allocation	Hybrid Spectrum Access Algorithm	None
A. Kaur, K. Kumar [31]	Dynamic spectrum management	Comparison-based Cooperative Q-Learning (CCopQL) and SARSA	Decentralized resource allocation

Authors	Treated problem	Processing in local	Scheme of cooperation
P. Yang et al. [32]	Dynamic spectrum access	Deep Q-learning	Balanced via replicator dynamic using evolutionary game theory
X. Tan et al. [34]	Distributed dynamic spectrum access	Deep recurrent Q-networks (DRQN)	Centralized offline training, distributed online execution
A. Gao et al. [35]	Cooperative spectrum sensing	Multi-agent deep deterministic policy gradient (MADDPG) algorithm	Centralized training, decentralized execution
A. Gao et al. [36]	Cooperative spectrum sensing	Attention mechanism for partner selection with MADDPG algorithm	Centralized training, decentralized execution
T.J. O'Shea et al. [38]	Radio modulation classification	CNN	None
N.E. West, T O'Shea [40]	Radio modulation recognition	Deep NN	None

Table 1. Overview or related work

1.3 Scientific Contribution

The Internet of Things (IoT) represents a rapidly evolving paradigm, often described as a global network of uniquely identifiable and interconnected objects. These objects leverage standard communication protocols to enable seamless connectivity between people and devices at any time, in any location, and across a broad range of platforms and services [41]. This vision of pervasive interconnectivity is expected to rely heavily on wireless communication, leading to an increased demand for communication resources, specifically frequency spectrum allocation.

As IoT continues to expand, the surge in the number of connected devices—both licensed and unlicensed spectrum users—will intensify the need for greater throughput and introduce challenges such as spectrum scarcity, interference, and mobility. These challenges underscore the necessity for IoT devices to incorporate cognitive capabilities. By integrating cognitive functionalities, IoT devices can gain the ability to perceive their environment, analyze available resources, and make intelligent decisions to optimize communication in dynamic conditions.

There are several application areas within the IoT realm that stand to benefit from the implementation of cognitive radio networks. These include, but are not limited to:

- smart grids and smart metering, in which CRNs can enhance energy management systems by enabling real-time consumption forecasting and demand response, improving efficiency and reliability;
- smart homes, where IoT technologies supported by CRNs can simplify device control, enable advanced monitoring, and facilitate content distribution while ensuring optimal network utilization and minimal interference;
- healthcare and medical applications, in which the integration of body area networks (BAN) with CRNs can support healthcare by providing continuous, real-time patient monitoring and treatment, ensuring better outcomes and reducing response times during emergencies;
- vehicular ad hoc networks (VANET), in which cognitive radio-equipped vehicular networks (CR-VANET) can support vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication, particularly in license-free radio bands, for the purposes of enhanced road safety, traffic management, and autonomous driving capabilities;
- emergency response networks, where in public safety and disaster management scenarios, ad hoc networks with cognitive capabilities could facilitate the transmission of video surveillance and sensory data to coordinate effective responses in real-time, even in challenging communication environments.

By enabling IoT devices to intelligently adapt to changing conditions, cognitive radio networks offer the potential to address key challenges in connectivity, paving the way for a smarter, more resilient IoT ecosystem.

CRN technology in standardized in certain areas of implementation, so in addition to applications in IoT technologies, CRN is seen as a regional wireless or WiFi technology that works in the white space" of TV signals (unused frequency and time resources) according to IEEE 802.22 or IEEE 802.11af standards, respectively. Standardization of dynamic spectrum access networks, such as CRN, is also addressed by the IEEE 1900 standards, which aim to facilitate the development of research ideas related to distributed cognitive radio networks into standards to accelerate the use of such ideas in commercial products.

The applications of reinforcement learning are definitely more diverse than those of CRN. Reinforcement Learning, especially the Deep reinforcement learning, which implements neural networks for the purpose of evaluating functions, has seen implementation in a relatively new area of the entertainment industry - video games. Multi-agent reinforcement learning (MARL) is proposed for applications in marketing and advertising, where the ability to precisely target an individual is crucial, as well as a trade-off between competition and cooperation among advertisers. RL is similarly proposed for recommendation systems. Applications and perspectives of RL in finance are also noted. Other areas of application of RL include natural language processing, healthcare, industrial automation, autonomous driving, etc.

The result, i.e., the outcome of the conducted research is that this rich and a very current area of research is richer with one more method of application of the MARL scheme, specifically in radio spectrum management within the example of CRN.

The work presented in the thesis introduces a novel distributed and decentralized algorithm for joint spectrum sensing and channel selection (JSS) in cognitive radio networks (CRNs). The proposed approach leverages multi-agent reinforcement learning (MARL) enhanced by distributed consensus iterations over directed, potentially time-varying communication graphs. The primary aim of this work is to demonstrate the efficacy of integrating consensus schemes within MARL frameworks to address key challenges in CRNs. Building on foundational ideas presented in earlier research [42], and as published in [43], this thesis proposes two algorithms for JSS, analyzing their theoretical properties and validating their performance through extensive simulations.

The thesis explores policy evaluation as well as policy optimization tasks in this context. For policy evaluation, a consensus mechanism is combined with the temporal difference (TD) learning algorithm [44], [45] in an off-policy setting. Here, a single target policy is evaluated while individual nodes (agents) implement distinct behavior policies tailored to their characteristics. For policy optimization, the consensus approach is applied to action-value function learning, where agents locally employ the Q-learning algorithm alongside an ε-greedy exploration/exploitation strategy. Each agent has its own set of preferred channels for exploration, enabling complementary exploration across the network. This collective strategy significantly accelerates convergence. Importantly, even if individual agents cannot independently evaluate or learn the optimal policy due to limited information or resources, the distributed consensus-based scheme enables the entire network to achieve optimal performance.

Agents communicate their local estimates of value functions, either state-value or action-value, exclusively with neighboring nodes via low-bandwidth communication links. This ensures a fully decentralized setup where each agent relies solely on its local observations and the information exchanged with its immediate neighbors, without requiring any central coordination. Despite this decentralization, after a sufficient number of iterations, all nodes in the network converge to a consistent estimate of the target value function. When Q-learning is applied, they collectively approximate the optimal policy. Additionally, the consensus process inherently introduces a denoising effect, as it aggregates and smooths local estimates through convexifications at each step. This feature enhances the robustness and accuracy of the learning process.

The proposed algorithm exhibits several key advantages. It is scalable, capable of handling large networks, and resilient to node or communication link failures. The convergence properties, assumptions, and limitations of the algorithm are rigorously analyzed and discussed. Simulation results underscore the benefits of this consensus-based MARL framework. Even with sparse neighbor-based communication graphs, the algorithm achieves performance that closely approximates the global optimum. Furthermore, numerical comparisons highlight the superiority of this approach, based on exchanging local value function estimates, over alternative cooperative reinforcement learning schemes [2], [31] that rely on sharing local rewards. The latter often suffer from higher variance in value function estimates, which reduces their efficiency.

The proposed platform can be seen as:

- A coordination tool for managing the actions of multiple nodes/agents, each responsible for distinct regions of the state space, yet working towards a common goal.
- A parallelization tool that enhances convergence speed, particularly useful for high-dimensional challenges, such as those involving a large number of frequency channels.
- A denoising tool that utilizes the ability to average varying noise realizations from different agents. This is especially advantageous when some agents face high error probabilities in spectrum sensing or encounter elevated Packet Error Rates (PER) on specific channels (e.g., due to fading or shadowing effects). In such cases, the decisions from these agents are typically corrected by a greater number of nodes with more favorable conditions.

In addition, part of the work on the thesis included the simulations of deep learning models for the purpose of modulation classification or recognition, for the purpose of implementation within CR network agents, where it can be used by the secondary users to recognize the known primary users of the spectrum resource. Several deep neural network architectures were explored and evaluated various, comparing their performance and demonstrating the potential of DL methods for directly learning temporal patterns from raw signal data. By applying the "blind" temporal learning, in which raw data samples are fed into the network, the models were trained on publicly available datasets [37] and the results were compared to some existing data [38], [40]. The work suggests combining the proposed consensus algorithm between agents running DL models can significantly improve the accuracy of modulation classification.

In terms of the dissertation itself, the set hypotheses have been proven.

This dissertation is structured to provide a comprehensive exploration of the application of distributed reinforcement and deep learning in cognitive radio networks. The **Introduction** presents the research problem, the objectives of the study, and the significance of the work. In the second section, **Cognitive Radio Networks**, the fundamental concepts and challenges of cognitive radio systems are discussed, along with their potential for improving spectrum utilization. The third section, **Reinforcement Learning**, introduces the principles of reinforcement learning, highlighting its applicability to dynamic decision-making in cognitive radio networks. Building on this, the fourth section, **Distributed Reinforcement Learning**, explores the concepts of decentralized learning and multi-agent cooperation, which are key to implementing spectrum sensing and channel selection in a distributed manner. In the **Neural Networks and Deep Learning section**, the dissertation covers the fundamentals of neural networks and deep learning techniques, emphasizing their role in enhancing the performance of cognitive radio systems. The sixth section, **Joint Spectrum Sensing and Channel Selection**, introduces the new algorithms based on consensus-based reinforcement learning for spectrum sensing and channel selection to improve the efficiency and reliability of cognitive radio networks. The seventh section, **Spectrum Sensing by Modulation Classification Using Deep Neural Networks**, focuses on the application of deep learning for accurate modulation classification in spectrum sensing, providing a robust method for identifying signal characteristics of the spectrum users in real-world environments. Finally, the **Conclusion** summarizes the findings, presents the contributions of the research, and suggests directions for future work in the field.

2. COGNITIVE RADIO NETWORKS

2.1 CRN Introduction

The word “telecommunicat[196] comes from Greek word: "tele" meaning "distant" or "far" and "communication" derived from the Latin word "communicare" meaning "to share" or "to make common". Ever since it has become feasible for a man to convey a message somewhere far away, regardless of the means employed, the need to be aware of all the agents and circumstances that effect one’s being and to be able to influence said circumstances has only been increasing. The perceived necessity felt by the modern-day individual to be fully immersed in all aspects of not only its own existence (social and technological aspects included), but the existence of all known, provided the motivation for an omnipresent network which would facilitate the connection of people and things at Any-time, in Anyplace, with Anything and Anyone. Khan et al. defined this [274]enomenon as Internet of Things (IoT). This network should ideally be established using Any path/network and Any service [46]. The ease-of-life requirement dictates that a significant portion of this communication, to the greatest extent possible, be performed wirelessly. It is only to be expected then, that the sheer volume of today’s estimated 15,7 billion interconnected devices (in the year of 2023), whether wide-area or short-range, not to mention the planned growth o[3] these numbers to an estimated 38,9 billion (in 2029) [47] will create an enormous demand for the communication medium, i.e., spectrum resource. The anticipated growth in the number of IoT devices operating as both licensed and unlicensed spectrum users, along with the rising demand for higher throughput and the interference challenges it entails, highlights the need to address their mobility and adapt to varying communication channels. Additionally, the inefficiencies associated with traditional radio frequency spectrum allocation serve as compelling incentives and driving factors for incorporating cognitive ca[339]ilities into IoT devices. These cognitive capabilities aim to equip IoT devices with the ability to make intelligent decisions.

The term Cognitive Radio was [199][96]ed by Joseph Mitola III [48] as an extension of the Software Defined Radio (SDR) with radio-domain model-based reasoning about RF bands, air interfaces, p[12]tocols and spatial and temporal patterns that moderate the use of radio spe[143]m. A cognitive radio can, therefore, be thought of as an intelligent, adaptive radio that can sense and dynamically adjust its operating parameters. A Cognitive Radio Network is an ensemble of such actors, communicating between and next to each other, in a shared radio frequency spectrum.

Cognitive Ra[260] Networks (CRN) have the potential to enhance various IoT applications. For instance, in smart grids and smart metering, CRN can improve processes such as consumption prediction. In smart homes, it can enable efficient control, monitoring, seamless configuration, and content sharing. In healthcare, Body Area Networks (BAN) supported by CRN can provide continuous real-tim[3] monitoring and aid in patient care. The transportation sector could benefit through CR-VANETs (Cognitive Radio-Vehicular Ad Hoc Networks), which facilitate communication betwee[3] vehicles and with infrastructure, even in crowded unlicensed spectrum bands. For public safety and disaster management, CRN can be utilized in emergency networks to support sensors and video surveillance in ad hoc setups, ensuring a timely and effective response. Additionally, CRN is also relevant for military applications [49].

2.2 Key Concepts in Cognitive Radio Networks

Based on the idea that the cognitive radio should be aware of its spectrum environment, identify the available resources therein and modify its own operational parameters in order to utilize the the thesis makes note of three essential concepts employed in CRNs. These are Spectrum Sensing, Dynamic Spectrum Access and Adaptive Transmission.

As it is conceived, one of the defining characteristics of a cognitive radio ts capacity for **Spectrum Sensing**. Performing this action enables a spectrum user in the form of a cognitive radi evice to detect the presence of other users, which could be primary, licensed users of the spectrum and t termine the availability of spectrum bands for its own (secondary, unlicensed) use. Techniques for spectrum sensing include energy detection, matched filtering, cyclostationary feature detection, radio identification based sensing and waveform based sensing [5].

Energy detection is a core technique in cognitive radio networks for efficient spectrum utilization. Unlike methods relying on known signal characteristics, it operates by measuring energy levels in frequency bands of interest. Receivers monitor power levels, and if they exceed a threshold, indicating primary user presence, the band is considered occupied. Its simplicity and versatility make it popular, but challenges include noise uncertainty and susceptibility to interference.

Spectrum sensing in cognitive radio networks often employs matched filtering, a technique t correlates received signals with known templates or waveforms. By comparing the received signal with the template, the presence of primary users can be detected. Matched filtering is particularly effective for detecting signals with known characteristics, such as those used by licensed users in specific frequency bands. This method offers high sensitivity and robustness against noise, making it sui e for detecting weak signals in noisy environments. However, matched filtering requires prior knowledge of the prima ser signals, which may not always be available or may need continuous updating to adapt t hanging conditions. Despite this limitation, matched filtering remains a valuable spectrum sensing technique in cognitive radio networks, especially in scenarios where signal characteristics are well-defined and stable.

Another form of sensing spectrum availability in cognitive radio networks is cyclostationary feature detection, which makes use the cyclostationary properties of signals to identify primary users. Unlike traditional methods, cyclostationary feature detection analyzes statistical properties, such as cyclic autocorrelation or spectral correlation functions, to detect periodicities indicative of primary user signals. This technique offers resilience against noise and interference and can detect signals with unknown or rapidly changing characteristics, making it suitable for dynamic radio environments. However, implementing cyclostationary feature detection requires complex signal processing algorithms and accurate estimation of cyclostationary parameters, posing challenges in practical implementation.

Radio identification-based spectrum sensing in cognitive radio networks involves identifying primary users by analyzing their unique radio frequency signatures. Each transmitter emits a distinct signature due to imperfections in its hardware components,

such as oscillators and filters. Cognitive radios use machine learning or signal processing to analyze received signals, extracting features like phase noise and modulation patterns for identification. By comparing received signals with stored signatures, cognitive radios can detect primary users without prior knowledge of their signals. This approach offers advantages in detecting unknown or changing signals but requires a comprehensive database and faces computational challenges.

Waveform-based spectrum sensing in cognitive radio networks involves examining the entire structure of received signals to detect and identify primary users. Instead of focusing solely on specific signal features, this method analyzes properties like modulation schemes and pulse shapes. By capturing and analyzing signals over time, cognitive radios can differentiate between primary users and other sources of interference. This approach is particularly beneficial in dynamic wireless environments where signals may have varying characteristics. However, it typically requires more computational resources and advanced signal processing techniques.

Spectrum sensing provides the foundation for **Dynamic Spectrum Access (DSA),** which can be formulated as the ability of cognitive radios to autonomously sense, identify, and exploit underutilized spectrum bands opportunistically in real time, while ensuring that licensed users are not subject to harmful interference. This dynamic allocation of resources is achieved through intelligent decision-making algorithms that consider various factors such as spectrum availability, quality of service requirements, interference mitigation and regulatory policies.

The concept of **Adaptive Transmission** in cognitive radios refers to the dynamic adjustment of transmission parameters based on the prevailing channel conditions, network requirements, and regulatory constraints, so as to utilize the identified available spectrum resources in the most efficient manner.

In CRNs, adaptive transmission techniques allow cognitive radios to adapt the characteristics of their transmission: transmit power, modulation scheme, coding rate, and channel bandwidth in response to changing environmental conditions. These conditions may include variations in channel quality, interference levels, user mobility, and spectrum availability.

Adaptive transmission techniques leverage information gathered through spectrum sensing, channel estimation, and feedback mechanisms to make informed decisions about transmission parameters. For example, if a cognitive radio detects deteriorating channel conditions or increased interference, it may adjust its transmit power or modulation scheme to maintain reliable communication while minimizing interference to other users.

Furthermore, adaptive transmission in CRNs also considers regulatory constraints and spectrum sharing policies. Cognitive radios must operate within allocated frequency bands and adhere to imposed power limitations. Adaptive transmission algorithms take into account these constraints to ensure compliance with regulatory requirements while maximizing spectrum utilization efficiency.

2.3 Cognitive Radio Network Architecture

The architecture of a CRN can be considered from the aspect of participating entities classification as well as the manner of their interconnection [50], [51]. The entities participating in a CRN are the primary network, which can be comprised of primary users (PUs) and primary base stations (Primary BS) and the secondary network, which can entail secondary users (SUs) and, possibly, secondary base stations.

Entities within the primary network are entities or systems that have priority access to specific portions of the radio frequency spectrum. These users are typically licensed by regulatory authorities and have exclusive rights to utilize designated frequency bands at a certain geographical location.

Entities within the secondary networks of CRNs are entities that opportunistically access spectrum bands licensed to the primary network, when those bands are not in use by the primary users. They, therefore, rely on dynamic spectrum access techniques to utilize spectrum resources efficiently and it is exactly for that reason that it is the entities from the secondary network that are equipped with cognitive capabilities.

From the aspect of entities interconnection, the secondary networks of CRN or the networks of actual cognitive radios, can be arranged in an infrastructure architecture [51], in which secondary users communicate with each other only through the secondary base stations or access points, or an ad hoc architecture, in which there are no secondary base stations or access points participating in the communication between the secondary users, or a mesh architecture, which is a combination of the two forementioned architectures in a manner which enables communication of secondary users either directly or via a base station or access point.
Regardless of the interconnection or entities within the primary and the secondary networks of the CRN ensemble, for the continuation of consideration within this work, we will consider all entities of primary networks as Primary Users (PUs), the owners of spectrum resources, and the entities within the secondary network as Secondary Users (SUs), which need to establish and perform communication within their network.

Part of the CR actually facilitating the DSA feature is known as the Cognitive Engine. It is the central decision-making system within the CR that enables the radio to adapt its operation dynamically to the surrounding radio environment. Leveraging ML, artificial intelligence (AI) or other decision-making frameworks, it manages spectrum resources to ensure optimal performance under varying conditions. Cognitive engine's components include the knowledge base, learning mechanism, optimization and reasoning modules and control interface. These components enable it to perform spectrum sensing, learning and decision making, resource management and policy compliance.

2.4 Cognitive Radio Applications

CRN can be applied across various domains which IoT technologies encompass [49].

In smart grids and smart metering, cognitive radios can be employed to provide

communication of interconnected agents in order to facilitate consumption forecasting and optimization of energy usage based on real-time data.

In smart home environments, CRN could streamline operations by enabling easy control, monitoring, and configuration of IoT devices. Additionally, it could aid in content distribution, ensuring seamless access to multimedia content across different devices within the home network.

In healthcare, integrating CRN with Body Area Networks (BAN) can enable continuous real-time patient monitoring and treatment, leading to enhanced healthcare efficiency and improved patient outcomes.

For Vehicular Ad Hoc Networks (VANETs), Cognitive Radio VANETs (CR-VANETs) can support communication between vehicles and infrastructure, even in crowded spectrum environments, improving connectivity and safety.

In emergency response systems, CRN offers significant advantages for public safety and disaster management by providing additional spectrum resources when needed. This capability supports seamless communication among surveillance cameras, sensors, and other devices in ad hoc networks, enhancing response effectiveness.

2.5 Standardizations

The principles of cognitive radios have been implemented in several IEEE standards in order to increase the efficiency of the subject communication.

IEEE 802.22 standard deals with Wireless Regional Area Networks (WRANs) which use white spaces in the television frequency spectrum and specifies the physical layer and cognitive medium access control layer of the air interface therein [52], while relying on precise GPS-based positioning and access to centralized databases of spatially and temporally available channels.

Cognitive radio capabilities are introduced to wireless metropolitan area networks by IEEE 802.16h standard (Cognitive WiMax) which defines said capabilities in all the bands in which IEEE 802.16 (WiMAX) standard is applicable, facilitating either the uncoordinated or the coordinated coexistence mechanisms between the networks operating at a certain region of interest [53].

IEEE 802.11af is another example of a standard that defines operations of participating devices in TV white spaces, making use of cognitive radio capabilities at the physical and medium access control layers, only for the Wireless Local Area Networks, with expected ranges of about a few kilometers. Hence its commercial name of White-Fi. As is the case with IEEE 802.22, the previously mentioned standard defining the operation in TV white space, only for regional area networks, White-Fi is also dependent on geolocation database and location servers for proper operation of participating entities [54].

In addition, the IEEE has formed standards committees for the development of

Dynamic Spectrum Access Networks (DySPAN) standards which issue the IEEE 1900 standards. These committees focus on enhancing spectrum utilization through dynamic spectrum access radio systems and networks. Their scope includes developing innovative methods for dynamic spectrum access, managing radio transmission interference, and enabling coordination among various wireless technologies. This also involves network management and facilitating information sharing across networks utilizing different wireless systems [53], [55].

3. REINFORCEMENT LEARNING

3.1 Machine Learning

Reinforcement learning is one of the ways that machines can be programmed to learn. Besides reinforcement learning, two other paradigms are broadly recognized as major ones within machine learning and these are supervised and unsupervised learning.

In supervised learning, a model is trained using a labeled dataset, where each input is paired with a corresponding target output. The objective is for the model to understand the relationship between inputs and outputs, enabling it to make accurate predictions on new, unseen data.

Unsupervised learning, on the other hand, involves training a model on data that lacks labels. The algorithm identifies patterns or structures within the data independently, with no explicit guidance. Common applications include tasks like clustering and dimensionality reduction.

Reinforcement learning (RL) is a branch of machine learning where an agent learns to make decisions by interacting with its environment. The learning process is guided by feedback in the form of rewards, which the agent receives as it navigates through the environment. The primary objective of RL is for the agent to develop a policy a strategy that maps states of the agent and the environment to desired actions in order to maximize a cumulative numerical reward over time.

Unlike supervised and unsupervised learning, reinforcement learning relies on a trial-and-error approach. The agent explores its environment and refines its actions based on feedback, which can be either rewards or the absence thereof. This iterative process enables the agent to progressively improve its decision-making capabilities.

3.2 Reinforcement Learning Concepts

Reinforcement learning (RL) introduces and deals with several concepts. One of the concepts within RL is the concept of an *agent*. The agent is the autonomous entity that learns to make decisions, through its interaction with the *environment*. In light of the subject matter of this thesis, an agent could be a single cognitive module within a cognitive radio.

The environment represents the external system with which the agent interacts. The agent must be able to sense the environment, at least to some degree, and it interacts with the environment by performing *actions* that modify the environment in some manner.

For this modification of the environment, the agent is awarded a *reward*. The reward presented by the environment to the agent is a scalar value used to indicate the appropriateness of a certain action with respect to the agent's objective or goal in the environment. The main objective of an agent is to maximize the cumulative reward over time. This means that the goal of reinforcement learning is defined or driven by

means of the granted and gained reward(s).

A particular configuration or situation in the environment at any given time, which encapsulates the relevant information needed for decision-making by the agent, is known as a *state* of the environment. An action of an agent can then be considered as a decision made at a particular state, which influences the subsequent state and potentially the reward received from the environment.

This sequence of events, where an agent's action alters the environment to reach a new state and results in a reward, is cyclical. The reward, reflecting the success of the agent's action, along with the updated environment state, forms the basis for the agent's next action. This process unfolds in stages of decision-making, which can be viewed as discrete time steps.

Collecting rewards through time, an agent can learn that some states of the environment are more favorable than others, as they provide for better long-term yield of rewards based on the sequence of states and actions that follow the one it is currently in. In order to measure the desirability of being in a certain state, an appropriate function is instated. A *value function* is therefore defined as a sum of rewards that can be expected to be gained in the future, starting from a certain state or action. Depending on the origin of accumulation of these rewards, the value function can defined for either a state, or an action taken from a certain state, thus defining the either the (State-) Value function or the Action-Value function. These functions guide the agent's decision-making process by helping it evaluate the desirability of different states or state-action combinations.

In order to gain a reliable estimate of a value or a certain state or action, each action must be tried numerous times. Once the values of some states and actions are known or estimated, in order to satisfy its objective (of maximizing the sum of rewards), an agent should favor either the states or actions with the highest value, i.e., it should act in a *greedy* manner and *exploit* the knowledge gained thus far. However, *exploiting* the states or actions of the highest values at each time step could mean that an agent might not be able to discover some other states or actions, of suboptimal estimated value at that particular point in time, which could lead to an even higher accumulation of rewards in the future than currently known to the agent. It makes sense, then, for an agent to *explore* the pool of states and actions, by choosing to reach a state or select an action which is not of the highest value at a certain time step, just for a chance to *learn* a better, more valued sequence of states and actions to the goal. It can be said, then, that exploration involves testing different actions to gain new insights into the environment, whereas exploitation focuses on selecting actions that are thought to be optimal based on existing knowledge. Striking the right balance between exploration and exploitation is essential for efficient learning in reinforcement learning.

A set of rules employed by the agent in order to establish its actions based on its current state is called a *policy*.

In order to optimize its behavior within its environment, an agent would use policy evaluation and improvement. In RL, these two processes play a critical role in enhancing the agent's performance.

Policy evaluation involves determining the value function for a given policy. The value function predicts the expected return or cumulative reward an agent can obtain by following a specific policy while interacting with the environment. Several methods can be used for policy evaluation, such as dynamic programming (e.g., iterative policy evaluation), Monte Carlo techniques, and temporal difference learning (such as TD(0) or TD(λ)). The goal of policy evaluation is to measure the effectiveness of a policy based on the expected future rewards. By accurately estimating the value function, the agent can make better-informed decisions to maximize its long-term reward.

Policy improvement, on the other hand, involves enhancing the agent's policy based on the estimated value function obtained from policy evaluation. The aim is to iteratively refine the policy to achieve better performance. One of the common approaches to policy improvement is the greedy policy improvement method. This method selects actions that maximize the expected return according to the current value function estimate. In other words, the agent selects the action with the highest estimated value in each state. However, as this can lead to suboptimal policies, another approach is to use stochastic policies, where the probability of selecting each action is based on its estimated value. This stochastic method allows for more exploration and can lead to learning of higher value policies. Policy improvement can be done iteratively, with the processes of policy evaluation and policy improvement taking turns one after the other until an optimal policy is reached. This process is often referred to as policy iteration.

3.2.1 Markov Decision Process

RL operates on the principle of learning through trial-and-error by interacting with an unknown environment. A mathematical framework particularly well-suited for modeling decision-making in scenarios where outcomes are partially random and partially influenced by the decision-maker is the Markov Decision Process (MDP). MDPs are foundational to understanding and applying reinforcement learning and solving sequential decision-making problems.

As needed in the RL setup, MDP deals with components such as states, actions, transition probabilities and reward. The concepts of states, actions and rewards have already been explained in previous text. The transition probability is a probability associated with each state-action pair, indicating the likelihood of transitioning to a new state given a particular action is taken in the current state.

MDPs are defined by the Markov property, which asserts that the future state depends only on the current state and action, independent of the system's past states and actions. In other words, the system's history before the present state is irrelevant for predicting future outcomes, given the current information.

In this thesis, we utilize MDPs to model the system underlying a typical RL problem setup. An MDP can be characterized by the tuple $(\mathcal{S}, \mathcal{A}, r, T)$, where:

- $\mathcal{S}$ represents a discrete set of possible states. The state of an agent at a specific discrete time step n is denoted by $s(n)$,
- $\mathcal{A}$ denotes a discrete set of available actions. The set of actions available to the agent when in state $s(n)$ is represented by $\mathcal{A}(s(n))$,
- $r: \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ defines the reward function, which provides an agent with a

numerical reward (or, in case of random rewards – an expected reward) for an action taken from a certain state. The reward received by the agent at time step n, while in state $s(n)$, performing action $a(n) \in \mathcal{A}(s(n))$ and transitioning to state $s(n)$, is denoted by $R(n)$,

- $T: \mathcal{S} \times \mathcal{A} \to \mathcal{S}$ describes the state transition function, which determines the next state $s(n+1)$ after the agent executes action $a(n) \in \mathcal{A}(s(n))$ in state $s(n)$. In nondeterministic environments, we define P is defined as a probability distribution over states and actions, i.e., $P: \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0,1]$.

Additionally, we define a policy function $\pi: \mathcal{S} \to \mathcal{A}$, which prescribes the action $a(n)$ to be taken when the agent is in state $s(n)$. This policy can also be represented probabilistically as $\pi: \mathcal{S} \times \mathcal{A} \to [0,1]$, in which case it is referred to as a randomized policy.

The purpose of RL for an agent can then be formulated as the discovery of the optimal policy π^*, which maximizes a predefined function of the rewards received from actions taken over time. To model the agent's level of impatience, future rewards may be discounted. The objective in RL of an agent can thus be formulated as:

$$\text{find } \pi^* \text{ maximizing } E_\pi \left[\sum_{n=0}^{\infty} \gamma^n R(n) \right], \tag{1}$$

where E_π represents the mathematical expectation with respect to the Markov chain induced by policy π, and $0 \leq \gamma \leq 1$ is the discount factor for future rewards. In case $\gamma = 0$, the agent prioritizes immediate rewards, while $\gamma = 1$ signifies an emphasis on the cumulative sum of all rewards, which is feasible only when there exists an absorbing state with zero reward.

To compute the optimal policy, two value functions are typically employed. The *state-value function*, $v^\pi(s)$, represents the expected reward when following policy π from state s. When the environment's model is known, it can be computed using the Bellman equation:

$$v^\pi(s) = \sum_{a \in \mathcal{A}(s)} \pi(s,a) \sum_{s' \in \mathcal{S}} P(s'|s,a) \times \left(r(s,a,s') + \gamma v^\pi(s') \right) \tag{2}$$

The *action-value function*, $Q^\pi(s,a)$, quantifies the expected reward when the agent executes action a from state s and subsequently follows policy π. It is computed as:

$$Q^\pi(s,a) = \sum_{s' \in \mathcal{S}} P(s'|s,a) \times \left(r(s,a,s') + \gamma v^\pi(s') \right). \tag{3}$$

For the optimal action-value function Q^* under the optimal policy π^*, the relationship is given by:

$$Q^*(s,a) = \sum_{s' \in \mathcal{S}} P(s'|s,a) \times \left(r(s,a,s') + \gamma \max_{a'} Q^*(s',a') \right). \tag{4}$$

In scenarios where the environment's model is not known (i.e., we do not know the reward and the state transition functions), model-free RL techniques are employed to evaluate or determine the optimal policy. Equation (2) serves as the foundation of the Temporal Difference (TD) learning algorithm, while Equation (4) underpins the Q-learning algorithm [9]. In this thesis, distributed variants of both these algorithms will be introduced.

3.2.2 Temporal-Difference Learning

Even though other learning methodologies exist, Temporal-difference (TD) learning is without a doubt the one mostly associated with RL. TD is a combination of Dynamic Programming (DP) and Monte Carlo methods, with which it can be combined in many ways.

Dynamic Programming (DP) is an approach to solving complex problems by dividing them into smaller, more manageable subproblems. Each subproblem is solved once, and its solution is stored to prevent repetitive calculations. This powerful technique is commonly used in optimization and decision-making problems. In the context of RL, DP is used to compute optimal policies or value functions for Markov decision processes (MDPs), where an agent makes sequential decisions in an environment, whose model is known, to maximize cumulative rewards.

Because of the requirement for a known model of the environment and its computational requirements, DP is of limited use in solving RL problems.

As indicated, DP can be used to calculate the state-value function $v^{\pi}(s)$ for an arbitrary policy π. This is known as the *policy evaluation* or *prediction problem* in DP. If model dynamics is known, Equation represents a system of $|\mathcal{S}|$ linear equations with $|\mathcal{S}|$ unknowns (unknowns being the values of states $v^{\pi}(s), s \in \mathcal{S}$) which can, in theory, be solved. Iterative process in which Equation (2) is used as an update rule to obtain the next successive approximation of the value of state s, $v(s')$, is known as *iterative policy evaluation* and it can be shown to converge to the values of $v^{\pi}(s)$.

The processes of *iterative policy evaluation* are used in the process of *policy iteration* for the purpose of computing the optimal policy by iteratively improving the current estimate based on the Bellman equations.

Q-learning is a widely used model-free, off-policy temporal difference learning algorithm designed to identify optimal policies in Markov Decision Processes (MDPs). The core concept of Q-learning involves an agent iteratively updating its Q-values (as described in Equation (3)) based on the rewards it obtains from the environment. This process is guided by the agent's exploration-exploitation strategy.

In large state and action spaces, Q-learning can become computationally expensive, as it requires maintaining and updating Q-values for each state-action pair. Because of this, schemes extending the original Q-learning algorithm have been developed. Schemes such as Deep Q-learning (DQN) integrate deep neural networks to approximate, rather than calculate, Q-values for high-dimensional state spaces. Examples of further modifications of this scheme are Double Q-learning, which implements two Q-value estimator networks in order to address the overestimation bias, or Deep Recurrent Q Networks (DRQN), which extends Q-learning by using recurrent neural networks (RNN) to maintain the memory of past states in order to tackle environments with partial observability.

4. DISTRIBUTED REINFORCEMENT LEARNING

4.1 Motivation and Approaches to Distributed Reinforcement Learning

Several challenges and limitations of traditional RL, particularly in the context of complex, large-scale, and resource-intensive environments have brought about the need for introduction of a distributed RL.

From the aspect of scalability for complex problems, traditional RL often struggles to scale to problems with large state and action spaces due to the computational and memory demands of training neural networks or other function approximators. By distributing the computational workload across multiple agents, nodes, or processors, distributed schemes can handle significantly larger and more complex tasks than traditional single-agent RL.

Distribution of RL tasks among agents also enables accelerated training through parallelization. Whereas traditional RL algorithms typically require a large number of interactions with the environment to learn effectively, resulting in slow convergence, particularly in environments with sparse rewards or long horizons, parallelization of tasks between agents facilitates simultaneous exploration of the environment, collection of experiences, and sharing them for training. This parallelization reduces training time significantly while improving sample efficiency. Distributed agents can explore different parts of the environment independently, leading to better coverage of the state space and reducing the likelihood of getting stuck in suboptimal policies.

Distributed systems also increase fault tolerance, as the failure of a single agent or node does not necessarily compromise the overall training process. Other agents can continue to contribute to the learning process.

Distributed RL encompasses several major approaches, each designed to leverage the power of distributed systems while addressing challenges of scaling RL algorithms. These approaches can vary based on how agents interact with each other and with the environment.

These approaches include parallelized RL, in which there is no interaction between agents, but multiple agents independently interact with the environment, collecting experiences in parallel, so as to increase sample efficiency and exploration. These experiences are then sent to a centralized learner for policy or value function updates. [56] [57]

Another approach is the game theory approach to RL, which is primarily concerned with scenarios involving multiple agents interacting within an environment, where each agent's actions influence not only their own outcomes but also those of other agents. In such settings, the dynamics can be modeled as a game, with reinforcement learning used to learn optimal strategies for each agent. This is often implemented in MARL. [58] [59] [60] [61] [62] [63]

Between these two extremes, so to say, lays the approach implemented in the thesis,

which is one of decentralized RL in which agent interaction is present, multiple agents operate independently and share knowledge or coordinate with other agents to optimize a global objective, interacting through experience sharing, policy sharing, or collective decision-making, but they do not influence each other in a competitive manner.

4.2 Distributed Temporal Difference Algorithm

In order to introduce the distributed versions of the reinforcement learning algorithms to be implemented, let us first delve into the theoretical treatise of the same [64].

Consider a collection of N MDPs, marked as MDP $i = 1, \dots, N$, each defined according to previous Chapter 3.2.1 by the tuple $Q = \{\mathcal{S}, \mathcal{A}, r(s,a,s'), P(s'|s,a)\}$. Here, $\mathcal{S} = (|S| = \dots$ and $\mathcal{A}$ represent the sets of states and actions, respectively, and are both finite. Here, $r(s,a,s')$ represents the reward expected for a single step, with its realizations $R_i(s,a,s')$ being random a depending on i, while the transition probability $P(s'|s,a)$ defines the likelihood of transitioning to state $s' \in \mathcal{S}$ from state $s \in \mathcal{S}$ under action $a \in \mathcal{A}$

$\mathrm{MDP}^{(i)}, i = 1, \dots, N$, the employed stationary behavior policies $\pi^{(i)}(a|s)$ result in marginal state processes $\{s_i(n)\}$ for $i = 1, \dots, N$, further forming Markov chains indexed by instances of tran ition times $n \geq 0$. The objective then is to perform an evaluation of performance for a given target policy π by observations of the sequence of states, actions, and (random) rewards across $\mathrm{MDP}^{(i)}, i = 1, \dots, N$.

To formalize this, we also define $\mathrm{MDP}^{(0)}$, characterized by the same tuple Q, where the induced Markov chain $\{s_0(n)\} = \{s_\pi(n)\}$ corresponds to the target policy $\pi^{(0)}(a|s) = \pi(a|s)$. This setup falls within the domain of distributed off-policy collaborative learning problems, as explored in [9] [65].

If $\gamma(s) \in [0,1]$, $s \in S$, are state dependent discount factors (we generalize the set with constant discount factor introduced in the previous section), the performance of a policy π is evaluated using the value function, commonly referred to as the expected discounted total rewards. This value function is defined by the equation

$$v_\pi(s) = E_s^\pi \left\{ r_\pi(s_\pi(0)) + \sum_{n=1}^{\infty} \prod_{j=1}^{n} \gamma(s_\pi(j)) r_\pi(s_\pi(n)) \right\} \qquad (5)$$

where $r_\pi(s)$ represents the expected reward at state $s \in \mathcal{S}$ under policy π and $E_s^\pi\{\cdot\}$ denotes the expectation with respect to the Markov chain $\{s_\pi(n)\}$ starting from the initial state $s_\pi(0) = s$.

Let P_π denote the transition matrix of the Markov chain $\{s_\pi(n)\}$ induced by policy π, and let Γ be the $M \times M$ diagonal matrix with diagonal entries $\gamma(s), s = s_1, \dots, s_M$.

Assumption A1 (Target and behavior policies' assumptions)

a) P_π is such that that $I - P_\pi$ is nonsingular.

b) Policies $\pi^{(i)}(s)$ induce irreducible Markov chains on $\mathcal{S} \times \mathcal{A}, i = 0, \ldots, N$. Furthermore, for all $(s, a) \in \mathcal{S} \times \mathcal{A}, \pi^{(i)}(a|s) > 0$ if $\pi(a|s) > 0, i = 1, \ldots, N$.

According to the theory of MDPs, $v_\pi(s)$ from Equation (5) is well-defined under assumption A1.a, and it uniquely satisfies the fundamental Bellman equation $v_\pi = r_\pi + P_\pi \Gamma v_\pi$, which can be expressed equivalently as $v_\pi = (I - P_\pi \Gamma)^{-1} r_\pi$, where $v_\pi = [v_\pi(s_1) \cdots v_\pi(s_M)]^T$ and $r_\pi = [r_\pi(s_1) \cdots r_\pi(s_M)]^T$ are column vectors representing the value function and the expected rewards at states $s_1, \ldots, s_M$, respectively.

Let $\phi: \mathcal{S} \to R^p$ be a mapping that assigns each state $s \in \mathcal{S}$ to a p-dimensional feature vector $\phi \in \mathcal{R}^p$, and let $\mathcal{L}_\phi$ denote the subspace spanned by ϕ. The objective is to identify a function of the form $v(s) = \phi(s)^T \theta$, which serves as an approximation of $v_\pi(s)$, where $\theta \in \mathcal{R}^p$ is the parameter vector to be estimated. If Φ represents the $M \times p$ matrix whose rows are the p-dimensional vectors $\phi(s)$, the task reduces to determining the vector function $v = \Phi\theta \approx v_\pi, v = [v(s_1), \ldots, v(s_M)]^T$.

Assumption A2: Column vectors of Φ are linearly independent.

The following text describes a distributed recursive algorithm based on consensus, derived from the local TD(λ) algorithm applied to individual agents, as proposed in [64].

We consider local state-dependent functions $\lambda_i: \mathcal{S} \to [0,1], i = 0, \ldots, N$ (referred to as λ -parameters) [9]. For $\mathrm{MDP}^{(0)}$ with $\lambda_0 = \lambda$, the associated Bellman equation is given by $v = r_\pi^\lambda + P_\pi^\lambda v$, where $v = v_\pi$ is the unique solution. Defining Λ as the $M \times M$ diagonal matrix with entries $\lambda(s), s \in \mathcal{S}$, the $M \times M$ substochastic matrix P_π^λ and the M-vector of expected discounted total rewards $r_\pi^\lambda \in \mathcal{R}^M$ are expressed as $P_\pi^\lambda = I - (I - P_\pi \Gamma \Lambda)^{-1}(I - P_\pi \Lambda)$, and $r_\pi^\lambda = (I - P_\pi \Gamma \Lambda)^{-1} r_\pi$.

For $\mathrm{MDP}^{(i)}, i = 1, \ldots, N$, similar relationships hold, replacing λ with λ_i and Λ with Λ_i [65] [66].

We further assume that nodes have access to local importance sampling ratios $\rho^{(i)}: \mathcal{S} \times \mathcal{A} \to \mathcal{R}_+$, defined as

$$\rho^{(i)}(s, a) = \pi(a|s)/\pi^{(i)}(a|s) \tag{6}$$

for the purposes of correction of discrepancies between $\pi^{(i)}$ and π.
Following the temporal difference learning methodology, based on the observed states $\{s_i(n)\}$, the sequences of eligibility trace vectors $\{e_i(n)\}$ are updated recursively as

$$e_i(n) = \lambda_i(n)\gamma_i(n)\rho_i(n-1)e_i(n-1) + M_i(n)\phi_i(n) \tag{7}$$

with $(e_i(0) = e_{i0})$, where $\lambda_i(n) = \lambda_i(s_i(n)), \gamma_i(n) = \gamma(s_i(n)), \rho_i(n) = \rho_i(s_i(n), a_i(n))$ and $\phi_i(n) = \phi(s_i(n))$.

When $M_i(n) = 1, \forall n$ the previously given corresponds to eligibility trace generation in standard TD(λ).

The distributed algorithm further proposes to learn a linear approximation of the state value function for a given policy π, under strict inter-agent state and action accessibility and inter-agent communication. The algorithm comprises two parts:

1. **Local updates** based on local observations:

$$\theta_i'(n) = \theta_i(n) + \alpha_i(n)e_i(n)\rho_i(n)[R_i(n) + \gamma_i(n+1)\phi_i(n+1)^T\theta_i(n) - \phi_i(n)^T\theta_i(n)] \quad (8)$$

   where $\theta_i(0) = \theta_{i0}, R_i(n) = R_i\big(s_i(n), a_i(n), s_i(n+1)\big)$ is the current local reward, and $\alpha_i(n) > 0$ is a step size.

2. **Consensus updates** through inter-agent communication:

$$\theta_i(n+1) = \sum_{j=1}^{N} c_{ij}(n)\theta_j'(n) \quad (9)$$

   where $c_{ij}(n)$ are random variables forming a row-stochastic matrix $C(n) = [c_{ij}(n)], i, j = 1, \dots, N$, which respects the adjacency structure of the communication graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$. Here, $\mathcal{N}$ represents the set of nodes, $\mathcal{E}$ the set of arcs, and the graph's adjacency matrix $A^{\mathcal{G}} = [A_{ij}^{\mathcal{G}}], i, j = 1, \dots, N$ has entries $A_{ij}^{\mathcal{G}} = 1$, if nodes i and j are connected, and $A_{ij}^{\mathcal{G}} = 0$ otherwise. The $C(n)$ matrix then contains zeros in all positions (i, j) where the adjacency matrix $A^{\mathcal{G}}$ also has zeros, and nonnegative values in the remaining positions. Moreover, $C(n)$ is row-stochastic, satisfying $\sum_{j=1}^{N} c_{ij}(n) = 1, i = 1, \dots, , \forall n \geq 0$.

Let $\mathcal{N}_i$ denote the set of in-neighbors of node i, which includes all nodes directly connected to node i via an incoming edge [67].

The proposed algorithm, referred to as DTD(λ) in [64] resembles "running consensus" algorithms used for distributed parameter estimation, detection, and state estimation [68].

The total computational effort, at the network level, scales linearly with the number of nodes N, assuming a fixed $|\mathcal{N}_i|$. Regarding communication, the number of messages exchanged by node i at a given time step equals the sum of its in-neighbors and out-neighbors, corresponding to the number of ones in the i-th row and i-th column of the adjacency matrix $A^{\mathcal{G}}$.

4.2.1 Convergence analysis

The analysis of convergence for the proposed scheme is conducted using the framework of weak convergence methodology.

Consider N independent local learning algorithms derived from Equations (8) and (9) with $C(n) = I$. Define further

$$h_i(\theta_i, \xi_i) = e_i \rho_i(s, a)[r(s, a, s') + \gamma(s')\phi(s')^T\theta_i - \phi(s)^T\theta_i] \quad (10)$$

where $\xi_i = (e_i, f_i, s, a, s') \in \Xi_i = \mathcal{R}^{p+1} \times \mathcal{S} \times \mathcal{A} \times \mathcal{S}$. From Equations (8) and (9), the following update rule is obtained:

$$\theta_i(n+1) = \theta_i(n) + \alpha_i(n)\big[h_i\big(\theta_i(n), \xi_i(n)\big) + e_i(n)\omega_i(n+1)\big] \quad (11)$$

where $\xi_i(n) = \big(e_i(n), f_i(n), s_i(n), a_i(n), s_i(n+1)\big)$, and the term $\omega_i(n+1) = \rho_i(n)\big[R_i(n) - r_i\big(s_i(n), a_i(n), s_i(n+1)\big)\big]$ represents a zero-mean noise component [65]
.

This set of N local algorithms, described by Equation (11), is designed to solve N projected versions of the local Bellman equations: $v = \Pi_i\left(r_\pi^{\lambda_i} + P_\pi^{\lambda_i}\right)v$, $v \in \mathcal{L}_\phi \Leftrightarrow$ $A_i\theta_i + b_i = 0$, where Π_i denotes a projection onto $\mathcal{L}_\phi$ with respect to a weighted Euclidean norm. For the DTD(λ) algorithm, the norm weights are defined by the M-vector d_i, whose elements $d_i(s)$ correspond to the steady-state probabilities for policy $\pi^{(i)}$.

Let $D_i = \text{diag}\{d_i\}$ and $D_i^\mu = \text{diag}\{\mu_i\}$. Then, for DTD($\lambda$) we have:

$$A_i = A_i^a = -\Phi^T D_i\left(I - P_\pi^{\lambda_i}\right)\Phi, \quad b_i = b_i^a = \Phi^T D_i r_\pi^{\lambda_i} \quad (12)$$

Thus, the dynamics:

$$\dot{\theta}_i = \bar{h}_i(\theta_i) = A_i\theta_i + b_i \quad (13)$$

can be proposed as a candidate for the local mean ordinary differential equation.

Following the approach presented in [68], the communication aspect of the algorithm is considered as follows.

Let $\mathcal{F}_n$ represent an increasing sequence of σ-fields such that $\mathcal{F}_n$ measures $\{X(k), k \leq n, C(k), k < n\}$.

In order to present the theorems that prove the convergence of the algorithm, further assumptions are made:

Assumption A3: There exists a scalar $\epsilon_0 > 0$ such that $c_{ii}(n) \geq \epsilon_0$. For $i \neq j$, either $c_{ij}(n) = 0$ or $c_{ij}(n) \geq \epsilon_0$.

Assumption A4: There exist a scalar $p_0 > 0$ and an integer n_0 such that $P_{\mathcal{F}_n}\{\text{agent } j \text{ communicates to agent } i \text{ over the interval } [n, n+n_0]\} \geq p_0$, for all n and for all $i, j = 1, \ldots, N, A_{\mathcal{G}}^{(i,j)} \neq 0$.

Assumption A5: The directed graph $\mathcal{G}$ is strongly connected, meaning that for all pairs (i, j), there exists a path from i to j for $i, j = 1, \dots, N$.

Assumption A6: The sequence $\{C(n)\}$ is independent of the processes within $MDP^{(i)}, i = 0, \dots, N$.

Assumption A7: There exists an $N \times N$ matrix $\bar{\Psi}$ such that $E\{|E_{\mathcal{F}_k}\{\Psi(n)\} - \bar{\Psi}|\} \to 0$ as $n - k \to \infty$, where $|\cdot|$ denotes the infinity norm.

$\bar{\Psi}$ takes the form $\bar{\Psi}^T = [\bar{\Psi} \cdots \bar{\Psi}]$, where $\bar{\Psi} = [\overline{\psi_1} \cdots \overline{\psi_N}]^T$ with $\sum_i \overline{\psi_i} = 1$.

Let $X(n) = [\theta_1(n)^T \cdots \theta_N(n)^T]^T$ and $X'(n) = [\theta_1'(n)^T \cdots \theta_N'(n)^T]^T$. The global representation of the algorithm can then be expressed as:

$$X'(n) = X(n) + \alpha(n)\, H\,(X(n), n), \quad X(n+1) = \big(C(n) \otimes I_p\big)\, X'(n) \tag{14}$$

where $X(0) = X_0$, $\otimes$ denotes the Kronecker product, and $\alpha(n) = \text{diag}\{\alpha_1(n), \dots, \alpha_N(n)\} \otimes I_p$, $H(X(n), n) = [H_1(X(n), n)^T \cdots H_N(X(n), n)^T]^T$, $H_i(X(n), n) = \big[h_i\big(\theta_i(n), \xi_i(n)\big) + e_i(n)\omega_i(n+1)\big]$ and also $\bar{H}(X) = [\bar{H}_1(X)^T \cdots \bar{H}_N(X)^T]^T$, $\bar{H}_i(X) = E_{\zeta_i}\{H_i(X(n), n) | X(n) = X\}$.

Assumption A8: The sequence $\{X(n)\}$ is tight. [69]

Assumption A8 is commonly used in weak convergence analyses. As noted in [69] (in Theorem 3.1 there), it is possible to implicitly assume $\{X(n)\}$ is tight without loss of generality, employing simple truncation techniques for the estimates. Here, Assumption A8 is explicitly introduced to focus on other critical aspects of the problem.

Assume $\alpha_i(n) = \alpha > 0$. Let n_α be a sequence diverging to infinity as $\alpha \to 0$, satisfying $\alpha^{\frac{1}{2}} n_\alpha \to 0$ as $\alpha \to 0$ (for further details, see [69]).

Define $X_0^\alpha = \big(\Psi(n_\alpha | 0) \otimes I_p\big) X_0 + \alpha \sum_{k=0}^{n_\alpha - 1} \big(\Psi(k) \otimes I_p\big) H(X(k), k)$.

For $t \in \mathcal{R}^+$, define $X^\alpha(\cdot)$ such that $X^\alpha(t) = X(n)$ for $t \in [(n - n_\alpha)\alpha, (n - n_\alpha + 1)\alpha)$.

The core of the convergence analysis for the proposed algorithm is presented through three theorems in [64].

Theorem 1 establishes that the proposed distributed algorithm weakly converges to consensus, as defined by the mean ordinary differential equation derived from the properties of the state-trace processes and the communication part of the algorithm. The proof primarily relies on the general results from [69] (in Theorem 3.1 there) and [68] while carefully addressing the specific characteristics of the proposed algorithm.

Theorem 2 examines the asymptotic behavior of the estimates, ensuring the validity of the results uniformly over large (continuous-time) t and small step sizes α_i. The analysis begins with a stability investigation, which introduces new approaches for

stabilizing distributed TD schemes that may include potentially unstable subsystems. Weak convergence to the limit points of the mean ODE is then demonstrated using the general results from [69] (Theorem 5.1 there).

Theorem 3 focuses on increasingly long segments of the parameter trajectories' tails as $\alpha \to 0$. It concludes that the probability of entire trajectory segments remaining within an arbitrarily small neighborhood of the ordinary differential equation limit value approaches 1.

Theorem 1 [64]

Let assumptions A1–A8 be satisfied. Consider $X(n)$ generated by Equations (8) and (9), with $\alpha_i(n) = \alpha > 0$. Define $X^{\alpha}(0)$ such that $\lim_{\alpha \to 0} X_0^{\alpha} = [\theta_0^T \cdots \theta_0^T]^T$. Then, $X^{\alpha}(\cdot)$ is tight in $D[0, \infty)$, the set of real-valued functions on $[0, \infty)$ that are right-continuous and have left-hand limits, using the Skorokhod topology [70]. Moreover, $X^{\alpha}(\cdot)$ converges weakly to the process $X(\cdot) = [\theta(\cdot)^T \cdots \theta(\cdot)^T]^T$, where $\theta(\cdot)$ satisfies the mean ordinary differential equation:

$$\dot{\theta} = \sum_{i=1}^{N} \overline{\psi_i} \bar{h}_i(\theta) \tag{15}$$

with the initial condition θ_0.

The weak convergence arguments employed in Theorem 1 primarily provide insights into the location and/or distribution of $X^{\alpha}(\cdot)$ for small α and for t within large (but bounded) intervals. The subsequent theorem addresses the asymptotic behavior of $X^{\alpha}(t_{\alpha} + \cdot)$ as $t_{\alpha} \to \infty$ when $\alpha \to 0$.

Assumption A9: There exists an index $i = i_a$ such that $\mathbf{1}^T \widetilde{A_{i_a}^a} \succ 0$, where $\widetilde{A_i^a} = D_i \left(I - P_{\pi}^{\lambda_i} \right)$, where $x \succ 0$ indicates that the vector x has all positive components, and $\mathbf{1} = [1 \cdots 1]^T$.

Theorem 2 [64]

If assumptions A1 – A9 hold, then there exist coefficients $\{\overline{\psi_i}\}_{i=1,\ldots,N}$ such that the matrix $\sum_{i=1}^{N} \overline{\psi_i} A_i^a$ is asymptotically stable. Furthermore, $X(n_{\alpha} + \cdot)$, generated by DTD(λ), converges weakly to the constant process $\bar{X} = \left[\overline{\theta^T} \cdots \overline{\theta^T}\right]^T$, where $\bar{\theta}$ satisfies

$$\sum_{i=1}^{N} \overline{\psi_i} (A_i^a \bar{\theta} + b_i^a) = 0 \tag{16}$$

The result of Theorem 2 for DTD(λ) holds significant practical importance, as it demonstrates that global distributed TD(λ) algorithms can be stabilized through appropriate network design, even in scenarios where some local DTD(λ) algorithms

are unstable. Here, network design refers to defining $C(n)$ in such a way that it yields the desired values of $\overline{\psi_i}, i = 1, \dots, N$ (see [68] for further details).

The following theorem examines the scenario where $n \to \infty$ as $\alpha \to 0$ and demonstrates that the probability of the entire trajectory segment remaining within an arbitrarily small neighborhood of $\bar{X} = [\bar{\theta}^T \cdots \bar{\theta}^T]^T$ approaches one. This finding serves as a compelling illustration of the effectiveness of weak convergence methods.

Theorem 3 [64]

Let the assumptions of Theorem 1 hold, and let $\{k_\alpha | \alpha > 0\}$ be a nondecreasing sequence of nonnegative integers such that $\alpha k_\alpha \to \infty$ as $\alpha \to 0$. Then, there exists a sequence $\{T_\alpha | \alpha > 0\}$, with $T_\alpha \to \infty$ as $\alpha \to 0$, such that for any $\delta > 0$,

$$\lim_{\alpha \to 0} \sup P\{X(n) \notin \mathcal{N}_\delta(\bar{X}), \text{for some } n \in [k_\alpha, k_\alpha + T_\alpha/\alpha]\} = 0, \tag{17}$$

where $\mathcal{N}_\delta(\cdot)$ represents the δ-neighborhood of the specified set.

It is both theoretically and practically valuable to compare in terms of asymptotic bias the result from Equation (16) with that obtained via on-policy single-agent estimation, which serves as a reference. In the single-agent case, the projected Bellman equation is given by $v = \Pi_{D_\pi}(r_\pi^\lambda + P_\pi^\lambda v)$, where Π_{D_π} denotes the projection onto the approximation space with respect to the Euclidean norm $|v|_{D_\pi}^2 = \sum_{s \in \mathcal{S}} d_\pi(s) v(s)^2$, and d_π represents the invariant probability distribution for the target policy π ($D_\pi = \text{diag}\{d_\pi\}$). This Bellman equation yields $\theta_\pi = -[A^\pi]^{-1} b^\pi$, where $A^\pi = -\Phi^T D_\pi (I - P_\pi^\lambda)$ and $b^\pi = \Phi^T D_\pi r_\pi^\lambda$.

The distance between $\bar{\theta}$ from Equation (16) and θ_π can, in general, be reduced by exploiting the flexibility in choosing d_i, λ_i and $\overline{\psi_i}$ (via network design [68]) for $i = 1, \dots, N$. In the special case where $\lambda_i = \lambda$, and d_i are fixed in advance for $i = 0, \dots, N$, the problem reduces to determining $\overline{\psi_i}$, $i = 1, \dots, N$, that minimizes some distance between d_π and $\sum_i \overline{\psi_i} d_i$ in the case of DTD(λ).

Regarding the asymptotic error covariance, the asymptotic properties of the normalized error $|X^\alpha(n) - \bar{X}|/\sqrt{\alpha}$ can be analyzed by following the methodology outlined in [69]. Starting from a suitable set of assumptions, it is possible to derive that as $t \to \infty$ and $\alpha \to 0$, an Itô stochastic differential equation (SDE) serves as a continuous-time model. Utilizing this SDE approach and the results from [69] [44] it can be shown that the proposed distributed algorithm has the potential to reduce the asymptotic error variance compared to the single-agent case.

The problem is inherently complex, given the flexibility in selecting behavior policies and convexification coefficients. Framing this as a multi-criteria optimization problem could lead to a practically efficient solution that also accounts for stability and acceptable bias.

Due to the flexibility in choosing convexification coefficients and behavior policies, the considered problem is inherently complex. Formulating it as a multi-criteria optimization problem could provide a practically efficient solution while ensuring stability and maintaining an acceptable level of bias.

5. NEURAL NETWORKS and DEEP LEARNING

Contemporary and emerging mobile networks produce vast amounts of diverse and varied data. Utilizing deep learning frameworks and algorithms, rather than machine learning (ML), is recommended because the effectiveness of deep learning improves substantially as training datasets increase in size. Additionally, training deep neural networks (NNs) with large datasets reduces the risk of model overfitting. In contrast, training neural networks with Stochastic Gradient Descent (SGD) requires only small batches of data at each step, which makes deep learning scalable for large datasets [15].

Contemporary mobile networks are generating not only vast amounts of data but also highly heterogeneous data, including network traffic, user behavior, sensor data, and more. This diverse data requires more advanced approaches, like deep learning, to extract meaningful insights effectively.

One of the newer advancements is the growing recognition that traditional ML methods may struggle with the complexity and scale of mobile network data. Deep learning, on the other hand, is proving to be more effective because it can handle larger datasets, learn from more complex patterns, and adapt to changing data more effectively.

A notable development is the focus on methods like Stochastic Gradient Descent (SGD). While SGD has been a traditional method for training NNs, its ability to scale with large datasets is becoming more critical as mobile networks evolve. Research is continuing to refine how SGD can be optimized to handle real-time, large-scale data from mobile networks, improving scalability without requiring massive computational resources.

The ability to train deep neural networks with large datasets to reduce overfitting is becoming a more refined strategy. New approaches in model architecture, regularization, and data augmentation techniques are helping in preventing overfitting while maintaining model accuracy, making deep learning more robust and applicable in mobile network environments.

Emerging mobile networks are increasingly relying on edge computing rather than relying solely on cloud-based processing. This enables faster decision-making, reduces latency, and reduces the need to transfer large amounts of data to the cloud. This shift in infrastructure design is influencing how deep learning models are trained and deployed, with research focused on optimizing deep learning models for edge devices.

Mobile networks are generating massive amounts of unlabeled or semi-labeled data, leading to a rise in unsupervised and semi-supervised deep learning techniques. These techniques can identify patterns within vast amounts of unstructured data, eliminating the need for costly and time-intensive manual labeling. Researchers are increasingly focusing on improving these models to handle vast amounts of data with minimal labeled input.

These developments represent the cutting edge of research in applying deep learning

to mobile network management and optimization, particularly as networks become more complex, data-rich, and demand more adaptive, scalable solutions.

The rapid advancement of smartphones and laptops has greatly accelerated the adoption of AI applications on these devices. While model training often occurs in the cloud or on edge devices through federated learning techniques—beyond the scope of our study—our primary focus is on the inference process. Real-world data, including video, audio, and acceleration, can be captured by devices such as cameras, microphones, and sensors, enabling mobile applications to interact meaningfully with their surroundings.

Mobile network data is inherently diverse and often noisy. Deep learning offers a significant advantage over traditional machine learning (ML) techniques in handling this complexity, primarily through its capacity to capture and model intricate dependencies within data. Its ability to automatically extract high-level features from raw data eliminates the need for extensive manual feature engineering, making it highly effective at managing the complexity and variability of mobile network data. This strength positions deep learning as an invaluable tool for tasks such as traffic prediction, anomaly detection, and resource management, where traditional ML methods may fall short in identifying subtle patterns or handling large data volumes. Labeling such data manually would demand substantial human resources and prove costly [17].

Another key benefit of using deep learning in mobile networking is its ability to leverage unlabeled data to uncover valuable patterns through unsupervised learning techniques. This is particularly important in the context of modern mobile systems, which often produce vast amounts of semi-labeled or unlabeled data. Deep learning algorithms can automatically detect underlying structures within this data, eliminating the need for costly and time-consuming manual labeling processes. This capability opens up opportunities for more efficient data analysis, enabling improved network optimization, user behavior prediction, and anomaly detection, even when labeled data is scarce [16].

5.1 Overview of Neural Networks

A neural network is a type of machine learning model inspired by the way biological neural networks (like the brain) process information. It is a computational model made up of layers of interconnected nodes (neurons) that aim to identify patterns and relationships in data.

The essential components of a Neural Networks are described below.

Neurons, also known as Nodes are the fundamental units of a neural network. Each neuron receives inputs, processes them (through a weighted sum and activation function), and passes the output to the next layer.

Neurons in a network are organized in **layers**. Layers can be thought of as groups of neurons which are not interconnected among themselves and serve to accept data from the input or previous layer, process said data and forward the processed data to their outputs. Layers are named in accordance to their position within the network. The

layers “visible” from outside of the neural network are the input and the output layers. All the other layers are known as hidden layers. The first layer, known as the Input Layer, accepts the data (features) to be processed. The last layer is known as the Output Layer and it produces the final result (e.g., classification or prediction). Between these two layers lie the Hidden Layers, which perform the delegated computations and extract features through weights and biases.

Each **connection** between neurons has an associated **weight**, representing the strength of the connection. Adjusting these weights during training is how the network learns.

In order to provide flexibility to the model, a **bias** can be added to the weighted sum before applying the neuron’s activation function.

The **activation function** is the element or a feature of a neuron that introduces non-linearity into the model. Input data is passed through the layers of the network. Each neuron computes the weighted sum of its inputs, adds bias and applies an activation function, and sends the output to the next layer. The final layer produces the output. The most common functions used as activation functions include:

a) sigmoid function, which outputs values between 0 and 1,
b) ReLU (Rectified Linear Unit) function, which replaces negative values with 0,
c) Tanh, which outputs values between -1 and 1, and
d) Softmax, which converts raw outputs into probabilities.

In order to assess the error between the predicted output and the actual target value of a network, a **Loss Function** is implemented. Minimizing the loss function is actually the primary task of a NN. The implemented loss function depends on the task at hand, but the most commonly used loss functions include Mean Squared Error (MSE) or Mean Absolute Percentage Error (MAPE) for regression tasks and Cross-Entropy Loss for classification tasks.

The network learns, or is trained, in processes of **Forward Propagation** and **Backward Propagation**, whereas the process of forward propagation generates predictions based on the network’s current state and the process of backward propagation fine-tunes the network by learning from its mistakes, optimizing the parameters to improve predictions.

In order to establish an outcome for a given input, in the process of forward propagation, input data passes through the network layer by layer. Each neuron applies weights, biases, and activation functions to compute intermediate outputs. The final output layer generates predictions.

The process of backward propagation starts with **Loss Calculation**. Loss, or an error between the achieved and the desired output of the forward propagation, is calculated using the mentioned loss function.

The gradient of the loss function is then computed with respect to each parameter in the network by computing the derivative of the loss function with respect to each weight and bias in the neural network. These gradients are then used to update the parameters of a network through an **Optimizer**, for the purpose of minimizing the error

in the next pass of inputs through the network. Popular algorithms for this parameter update include Gradient Descent, in which weights are adjusted based on the gradient of the loss function and Adam, which additionally takes into consideration the exponentially weighted average of the gradients.

The process of network learning is iterative, whereas the mentioned steps of forward and backward propagation are repeated for multiple **epochs** (iterations over the dataset) until the model converges (i.e., the loss stabilizes or reaches a satisfactory level).

The network and the associated learning process are characterized by their **hyperparameters**, which are configuration settings formulated before and stay fixed throughout the training.

From the NN architecture side, the hyperparameters that define the network can include: the number of layers (determining the depth of the network (e.g., shallow or deep networks)), number of neurons per layer, activation functions (defining the transformation applied to each layer's output), dropout rate (controlling the fraction of neurons to "drop out" during training to prevent overfitting), etc.

Training related hyperparameters include: learning rate, batch size, number of epochs, optimizer, momentum, etc.

There are other hyperparameters to a NN, depending on its type and task, and these include regularization hyperparameters, dataset-related hyperparameters and similar.

5.2 Deep Learning Basics

Deep Learning constitutes a fundamental approach within Artificial Intelligence (AI) and is specifically a subfield of Machine Learning (ML), a methodology that allows intelligent agents to enhance their performance on specific tasks through data and experiential learning [14]. ML has demonstrated itself as the only feasible AI framework capable of functioning effectively in intricate real-world scenarios, with explicit programming being supplanted by adaptive learning processes. Unlike classical ML, which predominantly relies on the manual crafting of task-specific features, Deep Learning automates the feature design process systematically. This paradigm facilitates the creation of algorithms with exceptional performance and adaptability, rooted in a hierarchical representation of the problem at hand, with increasing complexity and abstraction at successive levels.

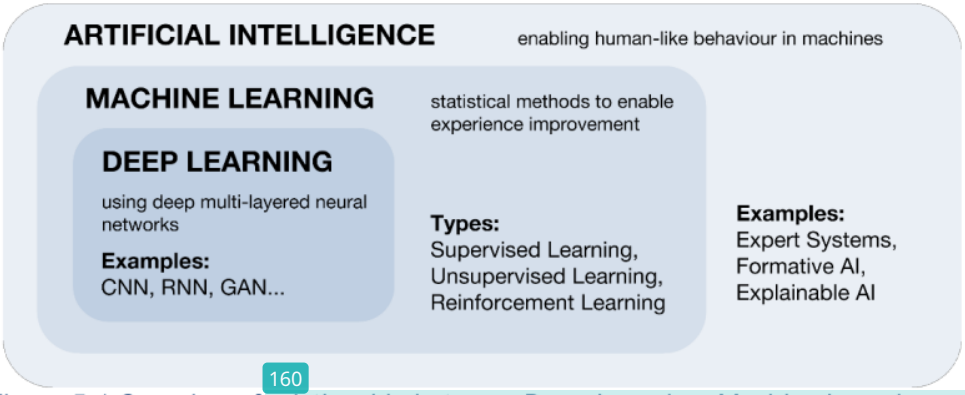


Figure 5-1 Overview of relationship between Deep Learning, Machine Learning and Artificial Intelligence

Machine Learning algorithms are broadly categorized based on the nature of the training experience they employ into three primary types: supervised learning, unsupervised learning, and reinforcement learning. Supervised learning involves training models on labeled datasets comprising input features paired with corresponding labels or targets to establish a desired mapping [9]. In contrast, unsupervised learning operates without labeled data, deriving meaningful structural properties directly from the data. Unsupervised learning within the Deep Learning context often involves learning the probability distribution underlying the input data. A hybrid approach, known as semi-supervised learning, combines labeled and unlabeled data within the same dataset. Reinforcement Learning (RL), on the other hand, does not rely on pre-existing datasets. Instead, it involves algorithms interacting with an environment, modifying its state, and receiving rewards for actions performed. The objective in RL is to devise a strategy that maximizes the cumulative expected reward during these interactions [9]. Application of Deep Learning models as function approximators in RL—known as Deep Reinforcement Learning (DRL)—has revitalized the field by enabling more sophisticated and scalable learning.

Feedforward deep neural network, often referred to as the Multi-Layer Perceptron (MLP), is a canonical example of a Deep Learning model. Inspired by biological neurons and synapses, MLPs consist of layers of interconnected neurons linked by weighted connections, where these weights represent trainable parameters. The training process adjusts these parameters to ensure the desired mapping of inputs to outputs [71]. The "deep" in deep neural networks signifies architectures with numerous hidden layers. Other multilayered architectures, such as deep random forests [72] or deep Gaussian processes [73], are sometimes classified within the broader Deep Learning framework, depending on the context.

MLPs utilize fully connected layers where each neuron is linked to all neurons in adjacent layers. Despite their historical popularity [74], fully connected layers are computationally intensive, particularly when applied to high-dimensional data like images. Convolutional Neural Networks (CNNs) address this limitation by introducing architectural constraints that reduce the number of parameters and optimize for spatially correlated data, such as images [75]. CNNs achieve this by limiting each neuron's connections to a localized subset of the input (the receptive field) and sharing weights among neurons through convolutional filters. These refinements have enabled

CNNs to achieve extraordinary performance in computer vision tasks, often surpassing human-level accuracy in image classification.

While contemporary Deep Learning often depends on large labeled datasets, alternative algorithms focus on unsupervised techniques and scenarios with limited data. A notable example is the Deep Belief Network (DBN) [76], composed of stacked Restricted Boltzmann Machines (RBMs), which are energy-based models with binary hidden and visible layers. The efficient training of DBNs through "greedy layer-wise pretraining" catalyzed a resurgence of interest in Deep Learning among AI researchers.

Auto-Encoders (AEs) represent another prominent unsupervised learning framework, designed to replicate inputs as outputs through a multi-layered structure, effectively reducing dimensionality in the process and learning compressed representations [77]. Variational Auto-Encoders (VAEs) extend AEs by incorporating probabilistic elements, enabling them to generate novel data samples resembling the input distribution [78].

Recurrent Neural Networks (RNNs) extend MLPs by introducing feedback loops, allowing their current states to depend on both the present inputs and prior states, thus enabling effective modeling of sequential data [79]. However, RNNs have struggled to capture long-term dependencies due to vanishing or exploding gradients during training.

Long Short-Term Memory (LSTM) networks address this challenge through specialized units designed to maintain gradient stability, complemented by gating mechanisms that regulate memory processes [80].

Gated Recurrent Units (GRUs), a streamlined alternative to LSTMs, achieve similar performance with reduced architectural complexity [81].

Recently, focus has shifted from RNNs to Transformer models [82], which rely heavily on the Attention mechanism [83] to weigh input data based on its contextual importance. Transformers are more parallelizable than RNNs, making them particularly suitable for large-scale datasets and computational efficiency.

Adversarial examples—inputs subtly modified to deceive models into confident but incorrect predictions—present significant challenges for many Deep Learning architectures [84]. This vulnerability has inspired the development of Generative Adversarial Networks (GANs). GANs involve two networks—a generator and a discriminator—engaged in a competitive, game-theoretic relationship. The generator creates synthetic data designed to appear authentic, while the discriminator aims to distinguish these from genuine data. The simultaneous training of these networks often poses significant challenges, as gradient-based methods are ill-suited to the inherently unstable, seesaw dynamics of GANs [85].

Notably, many of the architectures discussed can be adapted for supervised, unsupervised, or reinforcement learning applications, depending on the desired outcomes or the specific requirements of the implemented system. This flexibility underscores the transformative potential of Deep Learning across a diverse range of AI applications.

5.2.1 Reducing the Deep Learning Model's Complexity

Deep learning tasks often require large models and significant computational power. When deploying Mobile Deep Learning Applications (MDLAs) locally, challenges arise due to limitations such as insufficient memory or processing resources. A practical solution to this issue is model compression, which enables deep learning models to run efficiently on mobile devices. This compression technique has been widely adopted in deep neural networks (DNNs). Despite the structural differences among various DNN types, they typically consist of two main components: a feature extraction layer and a decision layer, each serving distinct functions.

The feature extraction layers are typically the primary contributors to computational costs, as they require extensive processing to handle input data. On the other hand, the decision layer, with its large number of parameters, contributes significantly to the model's storage requirements and overall size. In recent years, significant research has focused on model compression techniques, and this paper classifies these methods based on their space-memory (storage efficiency) and time-computation (computational efficiency) characteristics.

Reducing Model Spatial Complexity

Reducing the spatial complexity directly results in a significant reduction in memory requirements. There are two main strategies for achieving this: reducing the number of model parameters (via techniques like pruning and sharing) and compressing the model's size through methods like weight quantization. The need for optimized frameworks for deep learning on resource-constrained mobile devices has led to the development of specialized libraries that aim to address the limitations of traditional deep learning frameworks like TensorFlow, Caffe, and Torch. These general-purpose frameworks are designed for resource-rich environments and are not well-suited for mobile devices due to their heavy dependencies and focus on the training phase rather than the inference phase, which is critical for mobile applications.

Newer approaches focus on the optimization of deep learning models for inference on mobile devices. These include:

- Lightweight Frameworks: Tailored frameworks that prioritize the inference phase rather than training, such as TensorFlow Lite and PyTorch Mobile, are specifically designed to work efficiently on mobile devices. These frameworks reduce the size of models, use quantization to lower computational complexity, are optimized for mobile hardware.
- Model Compression and Acceleration: Techniques like model pruning, quantization, and knowledge distillation are used to compress models, reducing their size and computational demand. Mobile frameworks often implement hardware-specific optimizations, such as GPU or specialized AI accelerators, to maximize inference performance while minimizing resource usage.
- Mobile Hardware-Oriented Optimizations: Mobile deep learning frameworks are increasingly optimized to leverage mobile hardware, including GPUs, AI accelerators, and dedicated neural processing units (NPUs). These optimizations enable faster and more energy-efficient model inference, which is crucial for real-time applications.

- Efficient Data Handling: By minimizing the data transfer between the mobile device and cloud servers, some frameworks ensure that processing is done locally on the device, improving speed and preserving privacy. This localized processing is particularly useful in scenarios where privacy or bandwidth limitations are important considerations.

Research is also focusing on offloading specific tasks, such as object detection, to mobile devices while optimizing the process through federated learning or edge computing, allowing for the processing of large amounts of data in real-time on mobile devices without overburdening the device's resources. The focus is shifting towards lightweight, resource-efficient deep learning frameworks that optimize inference on mobile platforms. These frameworks aim to address the unique constraints of mobile devices while enabling complex, real-time applications like object detection, personalized health monitoring, and augmented reality.

A major advantage of deep learning is its ability to automatically extract high-level features from data with complex structures and intricate internal correlations. This process does not require manual design, greatly simplifying the need for prior feature engineering. This capability is particularly beneficial in mobile network contexts, where data often comes from diverse sources, tends to be noisy, and displays complex spatial and temporal patterns. Labeling such data manually would require substantial human effort.

Another strength of deep learning lies in its ability to manage vast datasets effectively. Mobile networks produce large volumes of varied data at a rapid rate. Training traditional machine learning algorithms, such as Support Vector Machines (SVM) and Gaussian Processes (GP), often requires all data to be loaded into memory, which becomes computationally impractical for big data scenarios. Additionally, traditional ML models often see diminishing returns as data volume increases, with performance gains plateauing relatively early. In contrast, deep learning—particularly with SGD for training—only requires data subsets at each step, ensuring scalability. Moreover, deep neural networks trained on large datasets benefit from reduced risks of overfitting, enhancing model robustness.

Traditional supervised learning relies on abundant labeled data to perform well, yet modern mobile systems typically generate mostly unlabeled or semi-labeled data. Deep learning, however, offers several techniques that can leverage this unlabeled data to identify meaningful patterns through unsupervised learning. Examples include Restricted Boltzmann Machines (RBM) and Generative Adversarial Networks (GANs), which have applications in clustering, modeling data distributions, and supporting unsupervised or semi-supervised learning. These methods also facilitate advanced techniques like one-shot and zero-shot learning, expanding their utility in scenarios with limited labeled data.

Deep learning excels at processing geometric mobile data, a challenge for many traditional machine learning techniques. Geometric data includes multivariate information defined by coordinates, topology, metrics, and order. Mobile data, such as user locations and network connections, can naturally be represented through point clouds and graphs that exhibit critical geometric features. Dedicated deep learning architectures like PointNet++ and Graph CNN effectively model these data types,

presenting substantial potential for transformative advances in analyzing geometric mobile data.

5.3 Deep Learning Architectures for Mobile Networks

Traditional mobile communication systems lack substantial computational resources necessary for supporting data-driven and deep learning applications. The next generation of mobile networks aims to address this limitation by adhering to key requirements, including: (a) ultra-low latency (under 1 ms); (b) exceptionally high communication speeds (>1 Gbps); (c) minimal energy consumption; and (d) enhanced reliability and service availability [86]. To fulfill the demands of Big Data and machine learning (ML) applications in fifth-generation (5G) cellular networks, innovative network architectures are indispensable.

The conventional service-driven architecture of 5G networks is predominantly cloud-based, where all services, including access, transport, core functionalities, and intelligent data analysis, are executed in remote data centers collectively referred to as the Cloud. These centralized facilities manage numerous services requiring significant computational resources while maintaining latency-aware resource control. However, for deep learning-based applications, such as augmented reality on mobile devices, cloud-centric data processing often fails to meet real-time requirements. Moreover, the simultaneous use of data-intensive applications across a vast number of devices (e.g., smartphones, IoT devices, tablets, wearables) can degrade the quality of service (QoS) and quality of experience (QoE) for end users, particularly in scenarios involving deep learning-powered video models [87].

To support intelligent applications powered by deep neural networks, distributed cloud paradigms like fog computing and mobile edge computing (MEC) have emerged as effective solutions. These paradigms tackle challenges related to real-time analysis, scalability, throughput, and energy efficiency by bringing computational processing, storage, and network resources closer to end users and data sources in mobile systems. Unlike the centralized cloud model, fog computing adopts an edge-centric framework that enables faster data processing and enhanced service quality in distributed cloud environments. Fog nodes, hosted on local servers closer to users than central cloud servers, extend the cloud paradigm by distributing heterogeneous resources and managing them in a decentralized manner. These nodes often act as filters or preprocessors for data destined for the cloud. This approach has been pivotal in enabling a range of intelligent services in next-generation mobile systems, such as anomaly detection [88].

MEC, on the other hand, plays a pivotal role in the integration of ML and Big Data applications within next-generation mobile networks. Unlike fog computing, which extends the network's edge, MEC situates computational resources directly at the network edge—closer to devices, such as base stations. This architecture, in conjunction with ML algorithms and software-defined networking (SDN), forms the foundation for deploying intelligent services in multi-tenant cellular networks. Deep learning models operating on MEC infrastructure have been applied to a range of challenges, including user count per base station [86], traffic prediction [89], channel state information estimation [90], network fault detection for low-latency communication [91], cache optimization to manage large data volumes [92], [93],

network traffic anomaly detection [94], handover prediction for improved user experience [95] and resource allocation [96].

Recent advancements have further enabled the deployment of deep learning models directly on end-user devices, referred to as mobile computing. While cloud, fog, and MEC paradigms rely on centralized or distributed resources, on-device machine learning processes data locally, offering advantages such as reduced response time, enhanced data privacy, and decreased communication bandwidth. However, mobile computing necessitates specialized hardware architectures, such as mobile GPUs and field-programmable gate arrays (FPGAs), integrated into devices like smartphones, laptops, and wearables.

Modern platforms supporting on-device deep learning include TensorFlow Lite, PyTorch Mobile, Mobile AI Compute Engine, Paddle Lite, and Core ML, among others. These frameworks empower devices to host applications in domains such as augmented reality, security, personal health monitoring, biometric analysis, and activity recognition [97]. By enabling AI capabilities directly on end devices, mobile computing positions these platforms as critical components for future machine learning endeavors.

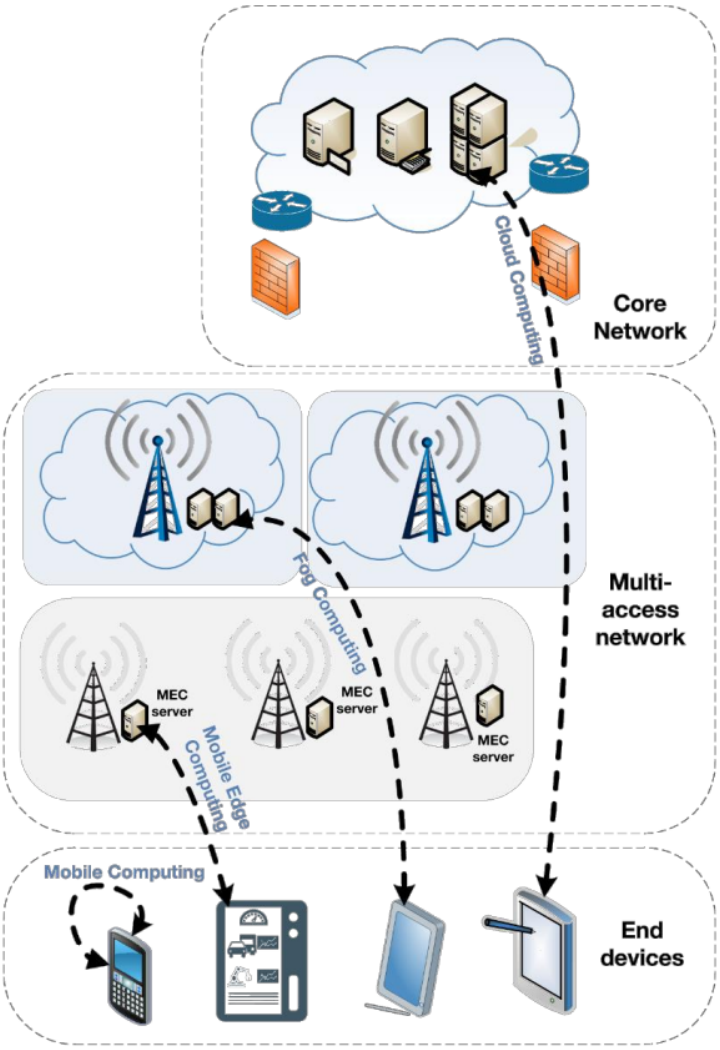


Figure 5-2 Overview of Cloud, Fog, Edge and Mobile computing

5.4 Deep Learning Applications in Mobile Networks

The rapid adoption of mobile devices and the growing popularity of mobile applications and services have created unprecedented demands on mobile and wireless networking infrastructure. As 5G systems evolve, they aim to support massive mobile traffic volumes, enable agile network resource management to optimize user experience, and provide fine-grained, real-time analytics. However, addressing these challenges is complex, as mobile environments continue to grow in diversity and dynamism. A promising solution lies in leveraging advanced machine learning techniques to manage the increasing data volumes and the proliferation of algorithm-driven applications. The recent success of deep learning offers powerful tools to tackle these challenges.

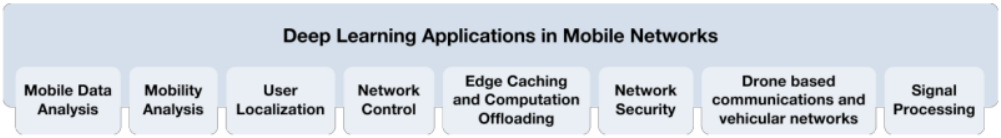


Figure 5-3 Deep Learning application subject areas in mobile networks

5.4.1 Mobile Data Analysis

Mobile networks produce data from a variety of sources, such as infrastructure metadata, performance metrics, call records, radio signals, end-user devices, user profiles, applications, and device sensors. As a major contributor to big data, this information is stored across cloud and edge-based systems. The study in [98] categorizes mobile network data into network-level and application-level, based on the origin of the data.

5.4.2 Network-Level Data

Network-level data exhibits spatial and temporal variations, influenced in part by the diversity of user behaviors [99]. This data proves valuable for analyzing user mobility, planning public transportation [100], and enhancing network diagnostics and management.

A key application of network-level data is in predicting network state, such as forecasting network traffic or estimating performance metrics based on historical data. For instance, high accuracy has been achieved in predicting Quality of Experience (QoE) using Multi-Layer Perceptrons (MLPs) with features such as average user throughput, cell activity, data volume per user, and channel quality indicators [101]. Additionally, deep learning approaches have proven effective for large-scale mobile traffic forecasting in urban areas, as they can leverage spatial and temporal correlations in traffic data. In one approach, Auto-Encoders (AEs) are used to extract spatial features and reduce dimensionality, while Long Short-Term Memory (LSTM) networks handle temporal dependencies, achieving more accurate forecasts compared to traditional models like Support Vector Machines (SVM) and Autoregressive Integrated Moving Average (ARIMA) [102]. Other studies ([103], [104], [105], [106]) have employed Convolutional Neural Networks (CNNs) and LSTMs for mobile traffic forecasting, consistently outperforming conventional models like ARIMA in accuracy.

Drawing inspiration from image super-resolution techniques [100], some researchers have explored the use of specialized Convolutional Neural Networks (CNNs) combined with Generative Adversarial Networks (GANs) to predict fine-resolution, network-wide mobile traffic from coarse-grained measurements. This approach aims to minimize measurement overhead, achieving up to 100-fold improvements in granularity when applied to real-world datasets. Another prominent research area for utilizing network-level data with deep learning is mobile traffic classification. In [107], deep Auto-Encoders (AEs) were employed to accurately identify protocols within TCP flow datasets. For encrypted traffic classification, studies in [108] applied MLPs, CNNs, and LSTMs, while [109] introduced a 1D CNN model, which is valued for its lower complexity. CNN architectures have also been used for malware traffic identification,

as shown in [110], where mobile data is treated as images, with representation learning used to classify anomalies indicative of malware.

A significant portion of network-level data is stored in the Call Data Register (CDR), and analyzing this data can provide valuable insights. Given their effectiveness in sequential data analysis, researchers in [111] utilized Recurrent Neural Networks (RNNs) to analyze user trajectories from CDR streams as location sequences, estimating metro density. Furthermore, CDR data was used in [112] with CNNs to predict user demographics like gender and age.

5.4.3 Application-Level Data

Application-level data is gathered from mobile devices, which act as sensor hubs for data collection and preprocessing [113]. This diverse data is sourced from GPS sensors, accelerometers, cameras, video recorders, and even wearable medical monitors, often collected through crowd-sourcing for analysis. It is intrinsically linked to user profiles, capturing behaviors such as mobility, preferences, and communication patterns [114]. Extracting useful patterns from this data can enhance recommender systems and enable targeted advertising; however, preserving user privacy while extracting meaningful insights from end-user sensors remains a significant challenge. Two primary approaches are used in application-level mobile data analysis: cloud-based and edge-based computing, distinguished by where computation occurs. In cloud-based computing, inference is performed in the cloud, and the results are transmitted to edge devices. In contrast, edge-based computing involves deploying models directly on edge devices to perform local processing.

One notable application of application-level data is in mobile health, where wearable sensors monitor users' physical conditions. For example, sensory devices can alert individuals with disabilities, such as those with hearing impairments, to emergency situations [115]. Other applications use deep learning networks to classify lifestyle data from wearables or support real-time health monitoring at the edge.

Application-level mobile data is also widely utilized in pattern recognition tasks, including object classification and activity recognition. For instance, some studies [116] explore the feasibility of implementing CNNs on mobile devices for object recognition. By comparing performance across CPU, GPU, and mobile platforms, the researchers demonstrate that such tasks can be effectively executed on mobile devices. In the area of activity recognition, research spans from analyzing offline smartphone datasets collected via accelerometers and gyroscopes to identify human activities [117], to using RBMs on smartwatch sensor data for activity analysis [118], and even to implementation of 4D CNN framework for processing of radio frequency signals with the purpose of designing a virtual "X-ray" machine which recognizes human poses [119].

Application-level mobile data also plays a crucial role in advancing Automatic Speech Recognition (ASR). In [120], a novel CNN architecture is proposed for ASR using mobile devices with multiple microphones, specifically designed to perform well in noisy environments. Another study [121] presents a deep learning framework for multi-task audio sensing on smartphones, addressing tasks such as speaker identification, emotion detection, stress recognition, and ambient scene analysis. This approach

delivers high accuracy while optimizing runtime, memory usage, and energy consumption.

5.4.4 Mobility Analysis

Patterns of individual and group movement, extracted from mobile network data, offer valuable insights for network resource management, public service optimization and urban planning. Deep learning, particularly its ability to capture spatial dependencies in sequential data, is highly effective for mobility analysis. In [122], LSTM networks are used to model city-wide movement patterns of people and vehicles. In [123], RNNs are applied to short-term mobility forecasting using real-world data, demonstrating higher accuracy compared to old methods.

5.4.5 User Localization

Location-based applications and services in mobile networks rely on precise user positioning data. An example of deep learning applied to this task is presented in [124], where crowd-sourced geo-tagged signal strength data is used to train an MLP, enabling it to learn the correlation between signal strength and geographic location.

5.4.6 Network Control

With deep learning's strong function approximation capabilities, advancements have been made in enhancing reinforcement and imitation learning, which has fueled forts to address complex network control problems. Alongside reinforcement and imitation learning, network control also employs analysis-based control. Reinforcement and imitation learning directly output actions, analysis-based control focuses on extracting valuable information to inform an agent's subsequent actions.

A key aspect of network control is network optimization, which focuses on efficiently managing network resources to improve overall performance. By leveraging mobile data analytics, operators can predict spatial and temporal variations in network resource demands [114]. This foresight enables more effective allocation of radio resources, ensuring that they are dynamically adjusted to meet the evolving needs of users, thereby enhancing network efficiency and user experience. In [125], researchers utilize LSTMs to forecast traffic load, adjusting resource allocation policies accordingly to prevent congestion. Additionally, [126] applies RNNs to predict traffic, aiding in the dynamic assignment of spectrum resources in mobile networks. Some studies have also explored network operations under challenging conditions. For instance, [127] tackles anti-jamming communication in unpredictable environments using a deep reinforcement learning agent. This approach integrates deep Q-networks with CNNs, using raw spectrum data as input, and significantly enhances network throughput while requiring minimal prior knowledge of the environment. This method allows for adaptive optimization of network performance, even in dynamic and uncertain conditions.

5.4.7 Network Security

Network security involves safeguarding network from unauthorized access. Modern cybersecurity systems are increasingly adopting deep learning techniques, both

through supervised and unsupervised learning methods. Deep learning is being leveraged for more advanced threat detection, anomaly identification, and predictive analysis. By training on vast amounts of network traffic data, deep learning models can identify patterns of malicious behavior, zero-day attacks, and even sophisticated, evolving threats that traditional security measures may miss. This integration of AI-driven security measures significantly enhances the ability to proactively defend against cyberattacks and mitigate potential risks in real-time.

Supervised learning can be applied to recognize unwanted patterns and signatures based on past experiences, allowing for their generalization to detect future intrusions. Unsupervised learning, on the other hand, helps identify patterns that deviate significantly from normal behavior. It is important to note that, beyond its use for protection, deep learning can also be leveraged from an attack perspective.

At the infrastructure level, research focuses on identifying network events that deviate from expected behavior, which could indicate a potential attack. Stacked Autoencoders (AEs) have been used to categorize network traffic with high accuracy [128], or in combination with Multi-Layer Perceptrons (MLPs) for feature selection and extraction [129]. AEs are also employed in [130] for detecting abnormal spectrum usage.

At the software level, security research primarily targets smartphone malware and botnets in modern mobile networks, utilizing deep learning techniques for their analysis and detection.

Restricted Boltzmann Machines (RBMs) have been used in [131] to classify Android malware by training on both labeled and unlabeled mobile applications. Other approaches for malware detection involve analyzing essential application features, where malware-related features are extracted using Deep Belief Networks (DBNs), and Support Vector Machines (SVMs) are used for classification. Convolutional Neural Networks (CNNs) are also applied in malware detection; in [132], CNNs detect malware by analyzing sequences of opcodes from disassembled byte-code of applications, treating the code as text for analysis.

Additionally, Multi-Layer Perceptrons (MLPs) or Long Short-Term Memory networks (LSTMs) are used for botnet detection, extracting features from mobile botnet behaviors to identify potential threats.
Security implementations of deep learning, particularly regarding user privacy, focus on ensuring the protection of private data during the training and evaluation of deep neural networks. Several studies have explored frameworks where deep learning tasks are partitioned between the cloud and mobile devices. In these frameworks, feature extraction is performed on mobile devices, while classification takes place in the cloud, enabling local processing of input data and providing strong privacy guarantees.

Deep learning networks are also researched in the context of cyber-attacks, aiming to compromise user privacy or predict sensitive information, such as passwords. Generative Adversarial Networks (GANs) are often employed for these purposes. In [133], various deep learning architectures are used to perform side-channel key recovery attacks. Compared to traditional machine learning methods, deep learning

chniques demonstrate superior performance in breaking both unprotected and protected implementations of the Advanced Encryption Standard (AES). Additionally, researc in [134] shows that Convolutional Neural Networks (CNNs) can accurately predict the list of active applications on a smartphone using only orientation data and information from a magnetometer. When motion sensor data is also incorporated, prediction accuracy can reach 98%, posing potential risks to user privacy. However, injecting Gaussian noise into the data has been shown to significantly reduce inference accuracy, helping mitigate privacy concerns.

5.4.8 Signal Processing

Deep learning techniques are increasingly being adopted in signal processing, particularly in applications like Multiple-Input Multiple-Output (MIMO) processing and modulation. These methods are proving effective in enhancing signal detection, improving system performance, and optimizing network resources, leveraging deep learning's ability to capture complex relationships within signal data.

In MIMO systems, deep learning aids in optimizing performance by adapting to varying environmental conditions. For example, in [135], MLP-based networks are explored as potential estimators for transmitted vectors in a MIMO channel. Simulations demonstrate that this architecture achieves high accuracy while requiring minimal computational resources. Additionally, for channel estimation and signal detection tasks, researchers have utilized both CNNs and MLPs, further highlighting the versatility and effectiveness of deep learning in these applications.

Researchers in [136] propose the use of non-iterative neural networks for controlling the transmit power of base stations, with the goal of reducing inter-cell interference. This approach takes advantage of deep learning models' efficiency to optimize power allocation without relying on iterative processes, enhancing network performance and minimizing signal interference between neighboring cells. The neural network estimates the optimal transmit power for each packet transmission, and simulations demonstrate that this method outperforms the current belief propagation algorithm.

In [137], deep learning is employed in design in the physical layer by modeling a single-user end-to-end MIMO system as an unsupervised deep autoencoder (AE). In this framework, the transmitter is implemented using an MLP followed by a normalization layer, while the receiver is constructed as another MLP, with the channel represented by a Gaussian noise layer. Experimental results demonstrate that this AE-based system significantly outperforms the conventional Space-Time Block Code method.

Tests evaluating deep learning platforms for modulation recognition indicate that Long Short-Term Memory (LSTM) networks are among the most effective models for this task, due to their ability to capture sequential dependencies in signal data. However, recent studies have shown that Convolutional Neural Networks (CNNs), typically used in image processing, can also achieve satisfactory accuracy in modulation classification. By extracting spatial hierarchies and patterns from the signal data, CNNs offer a robust alternative to LSTMs, particularly when dealing with complex, noisy environments or when training on large datasets. Both approaches represent significant advancements in signal processing for communications, contributing to more accurate and efficient modulation classification systems. While the benefits of

applying deep learning techniques across various mobile network domains have been emphasized, certain limitations still hinder its broader implementation. Deep learning is particularly susceptible to adversarial attacks, suffers from low interpretability, and demands large volumes of data for effective training.

Looking ahead, a promising direction for future research in mobile network applications of deep learning is the creation of large-scale, high-quality datasets to enhance network training. The availability of such comprehensive datasets will be crucial for improving the accuracy and generalization of deep learning models, enabling more effective network optimization, traffic prediction, and resource management. Additionally, these datasets will support the development of more robust models capable of handling the complexities and dynamics of modern mobile networks, paving the way for advanced applications in areas such as intelligent network control, anomaly detection, and user behavior prediction. With the growing significance of location-based services, advances in deep learning for spatio-temporal data mining are also anticipated [138]. Additionally, with the immense volume of unlabeled data generated by mobile networks, future research is likely to focus on unsupervised learning techniques aimed at automatically labeling and classifying this data. These methods could significantly enhance data processing and analysis, enabling more efficient network management. In the field of network control, continued advancements are anticipated in the application of deep reinforcement learning, which can optimize decision-making processes for dynamic resource allocation, interference management, and real-time network adjustments. This evolution will be key in addressing the complexities of modern network environments and ensuring optimal performance under varying conditions.

5.5 The Limitations of Application of Deep Learning

Despite the many advantages, there are some limitations in the application of DL.

Deep learning models, including deep reinforcement learning, have a well-documented vulnerability to adversarial examples—specially crafted inputs designed by attackers to mislead machine learning models. Such examples are often indistinguishable from legitimate data, yet they can cause significant misalignment within the model. Convolutional neural networks (CNNs), in particular, are susceptible to these attacks, potentially impacting deep learning's effectiveness in mobile systems. This vulnerability could be exploited in cyberattacks to bypass deep learning-based security measures. Developing deep models with robust defenses against adversarial examples is crucial, though it remains a challenging task.

Deep learning models, while achieving remarkable accuracy across various tasks, largely operate as "black boxes" with limited interpretability. Although they excel in performance, the underlying reasoning behind their decisions remains obscure, limiting their broader application in fields like network economics where understanding decision processes is essential. Consequently, many businesses prefer interpretable statistical models, even if this comes at the cost of accuracy. Recognizing this gap, researchers are actively working to enhance the transparency of deep learning models, making strides toward more interpretable solutions in the field.

Deep learning's success is highly data-dependent, often making data availability as crucial as the model itself. Training data augmentation can further enhance deep model performance, present[10] a unique opportunity within mobile networks, which generate vast data volumes. However, data collection can be costly[20]d raise privacy concerns, making it challenging to gather sufficient data for robust model training. In these cases, the advantages of deep learning may be outweighed by the logistical and ethical costs of data acquisition.

Deep neural networks often involve a large number of hyperparameters, and tuning them for optimal performance can be a challenging task. For example, c[10]figuring a single convolutional layer requires specifying hyperparameters such as the number, shape, stride, and dilation of filters, as well as the use of residual connections. As the model depth increases, the number of hyperparameters grows exponentially, significan[258]impacting the network's performance. Finding an optimal configuration can feel like searching for a needle in a haystack. AutoML platforms, such as those utilizing progressive neural architecture search, offer a potential solution to this issue. However, this process remains computationally expensive.

Deep learning demonstrates strong capabilities in processing geometric mobile data, which often poses challenges for traditional machine learning techniques[21]his type of data encompasses multivariate information characterized by spatial coordinates, topology, metrics, and sequential order. Examples of mobile data[21]with geometric properties include user locations and network connections, which can be naturally represented[10]rough point clouds and graphs, capturing essential geometric features. Specialized deep learning architectures, such as PointNet++ and Graph CNNs, are well-suited to model these data types, offering significant potential for breakthroughs in analyzing and leveraging geometric mobile data.

6. JOINT SPECTRUM SENSING AND CHANNEL SELECTION

In this chapter, two new algorithms for joint spectrum sensing and channel selection (JSS) in CRNs are presented. The algorithms are distributed and decentralized. For the purposes of consensus iteration in a MARL setting, algorithms employ operation through communication graphs which are directed and can be time-varying. The work aims to demonstrate the value of combining consensus mechanisms within MARL frameworks to address the unique challenges in CRNs. This study is the first to investigate consensus-based MARL approaches specifically in the context of CRNs.

Off-policy learning, where an agent evaluates a policy different from the one used during learning, is particularly relevant in distributed multi-agent settings [10][13], [139], [64]. In these scenarios, multiple agents operate and learn in parallel. Multi-agent reinforcement learning (MARL) has become a popular approach for decentralized and distributed applications.

A decentralized MARL framework allows agents to communicate intermittently through a network without relying on interactions governed by the underlying MDP [64], [140]. This decentralized approach eliminates the need for a central coordinator or fusion center. Recent developments in MARL have demonstrated robust theoretical support for its convergence and performance stability, making it a promising strategy for scalable, distributed decision-making.

For policy evaluation, we integrate consensus mechanisms with temporal difference (TD) learning in an off-policy context [6] , [45], allowing each network node to evaluate a single target policy while employing behavior policies tailored to their characteristics. During policy optimization, we introduce a consensus approach to action-value learning using an ε-greedy strategy, where each agent's unique channel preferences drive exploration. This setup promotes complementary exploration among agents, enabling effective policy evaluation and learning that would not be possible for individual nodes alone.

In this distributed scheme, nodes share local value function estimates with immediate neighbors via low-bandwidth links, enabling a decentralized structure where agents rely solely on local information and neighbor communications to make decisions, without requiring central coordination. Despite its decentralized nature, the algorithm ensures convergence: with sufficient iterations, all nodes approximate a consensus on the value function and, in the case of Q-learning, an optimal policy estimate. Furthermore, the consensus step introduces convexification, enhancing system stability and resilience to node and link failures.

We rigorously analyzed the algorithm's convergence properties, assumptions, and limitations. Extensive simulations demonstrate its effectiveness, showing that even with minimal neighbor communication, the algorithm's performance approaches that of a globally optimal solution [2] [31].

6.1 Problem Description and System Model

The model of the system and the description of the considered problem will be presented through an examination of the JSS task and the formulation of the JSS problem within the MDP framework.

6.1.1 Joint Spectrum Sensing and Channel Selection

Spectrum sensing is the first task required of SUs to identify the spectrum resources available for their own use. In real-world scenarios, developing cooperative schemes that combine information on channel sensing gathered from multiple SUs while relying on minimal, neighbor-based communication is highly beneficial. The step that follows is the dynamic channel selection (DCS), where all SUs participating in a CRN select spectrum dynamically, aiming to minimize interference with primary spectrum owners while maximizing their own performance. Together, these tasks form components of the JSS problem. These tasks must be executed continuously and with minimal delay at each node to adapt to the constantly changing environment. Leveraging effective cooperation is crucial to achieving this objective.

In the following, we describe the adopted CRN system model, which is widely used in existing literature on RL-based CRNs (e.g., Reference [2]). We consider a scenario with N SUs operating within the same sensing domain, forming a CRN. Each cognitive radio (CR), representing an SU, is equipped with a Dynamic Spectrum Access (DSA) module enabling environmental observation and the ability to change the operative frequency or channel, which serves as the primary transmission parameter in the adopted setup. The licensed band, shared by all CRN users (primary and secondary), consists of K frequencies $F = \{f_1, f_2, \dots, f_j, \dots, f_K\}$. Additionally, the model incorporates a low-bandwidth communication channel in the unlicensed band, designated for secondary user control signaling communication (CSC). Payload data packets are transmitted over the K licensed channels between SU transmitter-receiver pairs, denoted SU_i (transmitter) and RSU_i (receiver).

Primary user (PU) traffic on each of the K channels is modeled as a two-state birth-death process, with death rate α_j and birth rate β_j for channel j. Under this model, PU activity transitions (e.g., ON to OFF) follow a Poisson process with exponentially distributed inter-arrival times. Consequently, the posterior probabilities of channel occupancy by PUs can be expressed as:

$$p_j^{PU_{on}} = \beta_j/(\alpha_j + \beta_j),\ p_j^{PU_{off}} = \alpha_j/(\alpha_j + \beta_j),, \tag{18}$$

where $p_j^{PU_{on}}$ and $p_j^{PU_{off}}$ denote the probabilities of channel j being occupied (PU active) or vacant (PU idle), respectively [141]. Each channel also has an associated packet error rate (PER), denoted $\mathrm{PER_j}$.

SUs can perform one of three actions per time slot:

- SENSE: The SU senses the frequency it is tuned to, determining PU activity using a default energy-detection scheme [5] with a probability of correct detection p_D and error probability $1 - p_D$.

- TRANSMIT: The SU attempts to transmit a packet to RSU_i using Carrier Sense Multiple Access (CSMA) as the Media Access Control (MAC) protocol. Transmission continues until an acknowledgment packet is received. Alternatively, transmission ends when an allowed number of attempts MAX_ATT is exceeded, after which the packet gets discarded.
- SWITCH: The SU switches to a different licensed frequency and notifies RSU_i via the CSC channel.

The sequence of time slots for $\mathrm{SU_i}$ can be represented as $T_i = (\tau_i^0, \tau_i^1, \tau_i^2, \dots)$, where $\tau_i^n \in \{\mathrm{SENSE,TRANSMIT,SWITCH}\}$, based on the action taken in the n-th time slot. Decision-making by SU_i adheres to the following constraints:

1. If $\tau_i^n = \mathrm{SENSE}$ and the channel is busy, then $\tau_i^{n+1} \neq \mathrm{TRANSMIT}$ (i.e., the SU cannot transmit on a channel occupied by a PU).
2. If $\tau_i^n = \mathrm{SWITCH}$, then $\tau_i^{n+1} = \mathrm{SENSE}$, ensuring the SU assesses the occupancy of a newly selected channel before attempting transmission.

The model tracks transmission outcomes, including the total numbers of transmissions, successful transmissions, collisions with PUs, and failures due to other channel errors. With this model, the objective is to determine the optimal transmission schedule T_i for each $SU_i, i = 1, \dots, N$.

6.1.2 MDP Formulation of the JSS model

The former is sufficient to formulate the previously presented Joint Spectrum Sensing (JSS) model using the appropriate MDP formulations from the considerations in Chapters 3 and 4, which are well-suited for the application of reinforcement learning (RL) algorithms:

- The learning agents in the system are the entities SU_i, where $i = 1, \dots, N$.
- The current state of an agent SU_i, at a discrete time step n is denoted as $s_i(n)$, for $i = 1, \dots N$.
- The set of states, $\mathcal{S}$, consists of pairs (f_j, AVAIL_j), where f_j is a frequency from the set F (comprising K licensed frequencies or channels), and $\mathrm{AVAIL}_j = \{\mathrm{IDLE,BUSY,UNKNOWN}\}$ represents the sensed availability status of channel j.
- The action set for an agent SU_i, is given by $A_i = \{\mathrm{SENSE,TRANSMIT,SWITCH}_{f_1}, \dots, \mathrm{SWITCH}_{f_j}, \dots, \mathrm{SWITCH}_{f_K}\}$, for $i = 1, \dots, N$ and $j = 1, \dots, K$, where SWITCH_{f_j} represents the action of switching to frequency f_j, with the restriction that the agent does not switch to the frequency it is already tuned to.
- The reward function, $\mathcal{R}: \mathcal{S} \times \mathcal{A} \rightarrow \mathrm{PMF}[0,1]$, assigns rewards based on the actions performed and the states encountered, where $\mathrm{PMF}[0,1]$ denotes a probability mass function. The reward function is defined as follows:
  - $\mathcal{R}\left((f_j, *), \mathrm{SENSE}\right) = \zeta$, if f_j is found to be IDLE.

- $\mathcal{R}\left((f_j,*),\text{SENSE}\right) = 0$, if f_j is found to be BUSY.
- $\mathcal{R}\left((f_j,\text{IDLE}),\text{TRANSMIT}\right) = 1 - \frac{\#retrans}{MAX\_ATT}$, where $\#retrans$ is the number of packet retransmissions, a random variable dependent on the packet error rate (PER) and MAX_ATT is the already mentioned parameter of maximal allowed number of attempts of transmission.
- $\mathcal{R}\left((f_j,\text{UNKNOWN}),\text{SWITCH}\right) = 0$.

Here, ζ is a scalar parameter associated with the design of the model, as detailed in later chapter related to simulations.

- The state transition function, $\mathcal{T}: \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow [0,1]$, defines the probabilities of transitioning between states based on actions taken. It is specified as follows:
  - $\mathcal{T}\left((f_j,*),\text{SENSE},(f_j,\text{IDLE})\right) = \alpha_j/(\alpha_j + \beta_j)$
  - $\mathcal{T}\left((f_j,*),\text{SENSE},(f_j,\text{BUSY})\right) = \beta_j/(\alpha_j + \beta_j)$
  - $\mathcal{T}\left((f_j,\text{IDLE}),\text{TRANSMIT},(f_j,\text{IDLE})\right) = 1$
  - $\mathcal{T}\left((f_j,*),\text{SWITCH},(f_k,\text{UNKNOWN})\right) = 1$

  The state transition function assigns a value of 0 to all other combinations of arguments.

  It is important to note that, in practical scenarios, the transitions of channel states (e.g., from IDLE to BUSY or BUSY to IDLE) typically occur at a much lower frequency than the learning rate of the SUs. During such intervals, the probabilities of sensing IDLE or BUSY states can be set as either 0 or 1, as appropriate. Further details regarding this assumption are provided in the simulations section.

Figure 6-1 provides the MDP diagram for the JSS problem, highlighting states and actions specifically associated with frequencies f_i and f_j from the available set of K frequencies.

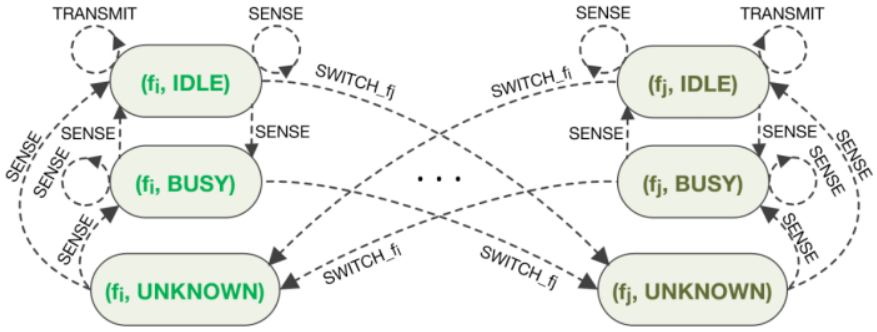


Figure 6-1 JSS formulation [43]

6.2 Consensus-Based Distributed Joint Spectrum Sensing and Selection

The following chapter describes novel, fully decentralized and distributed networked MARL solutions for addressing the JSS problem in CRNs. These solutions harness the potential of cooperative interaction between neighbor nodes while maintaining robustness and scalability. Building on adaptations and extensions of previously established MARL schemes (such as [64], [41], [141]), these approaches are tailored to suit the specific demands of JSS. This decentralized framework can be viewed through three key facets:

- Coordination of Multiple Agents: This approach enables nodes to coordinate their actions, covering different parts of the state space while working towards a shared objective.
- Parallelization for Faster Convergence: By parallelizing agent actions, the method achieves quicker convergence - especially useful in high-dimensional problems, such as those with numerous frequency channels.
- Noise Reduction through Agent Averaging: The decentralized structure allows the network to mitigate noise effects by averaging diverse noise realizations from individual agents. This is particularly beneficial in cases where some agents experience high error probabilities in spectrum sensing or face elevated PER on some channels due to variety factors like fading and shadowing. In such cases, the collective input from nodes with more favorable conditions can help offset inaccuracies from nodes experiencing higher error rates.

Even though based on the considerations from Chapters 3 and 4, the algorithms presented here differ somewhat from the DTD(λ) presented in Chapter 4 in two ways:

- In contrast to the formerly presented, these algorithms do not include parametrization, but rather aim to learn the actual values of either the (state-)value or action-value functions and
- the algorithms presented here are a special case of the former, in which $\lambda = 0$, i.e., updates to the value function rely solely on the immediate temporal difference error, and no credit is given to multi-step returns.

MARL framework is designed for a setup where N SUs (nodes or agents) are linked via a low-bandwidth communication network (in function or not of CSC). Here, each node can exchange information in real-time only with its immediate neighbors, a characteristic common in mobile ad hoc and sensor networks [6], [26], [143], [144]. This dedicated network is represented by a directed graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ where $\mathcal{N} = 1, \ldots, N$ is the set of nodes, and $\mathcal{E} = (i, j)$ the set of directed links (i, j). The neighboring nodes of node i, denoted $\mathcal{N}_i$ can communicate directly with node i, with $|\mathcal{N}_i| \ll N$ expected in large-scale networks.

Each SU_i operates within the MDP framework previously described, executing local policies, performing actions (such as sensing, transmitting, or switching channels), observing environmental responses, and receiving rewards that reflect the PU activity and channel quality. Notably, agents operate independently in this MDP structure, with transitions determined by individual actions, not direct inter-agent interactions. This independence can be maintained by a suitable distribution of channels, ensuring minimal interference among SUs.

We choose time instances k as the union of all T_i time instances, during which the secondary users (SUs) sequentially make decisions and trigger transitions within the MDP.

6.2.1 Distributed Consensus-based Policy Evaluation

This subsection addresses distributed policy evaluation within the multi-agent system described above. Here, each agent SU_i follows a distinct behavior policy π_i, making each associated MDP for SU_i a Markov chain (MC) with its own state transition matrix $P^{(i)}$, derived from function $\mathcal{T}$ by fixing policy π_i. Our focus is on the decentralized evaluation of a shared target policy π, which establishes a MC represented by the matrix P. The value function for π is given by Equation (2), and each agent thus seeks to estimate the vector $V^\pi \in \mathcal{R}^{3K}$, where 3K denotes the total number of states in our model. For this evaluation to work effectively, we assume the MC under target policy π is irreducible, meaning it has a limiting distribution $D \in \mathcal{R}^{3K}$and a unique value function V^π. Under policy π, each agent should ideally "visit" each channel infinitely often.

To account for differences between each agent's behavior π_i and the target π we introduce local importance sampling ratios (according to Chapter 4.2 and Equation (6)):
$\rho_i(s,s') = P_{ss'}/P^{(i)}_{ss'}$ for all $s,s' \in \mathcal{S}$ (with $0/0 = 0$), where $P_{ss'}$ and $P^{(i)}_{ss'}$ are the probabilities of transiting from state s to s' under π and π_i.

Each agent maintains its own estimate of the value function vector, denoted $V_i \in \mathcal{R}^{3K}$. For ease of reference, we define the global vector comprising all agents' estimates as $V_G = [V_1^T \cdots V_N^T]^T$.

Building on the methodology presented in [44] and Chapter 4, we frame a global constrained minimization problem across the network to measure how well the agents' value function estimates align with the Bellman equation:

$$J(V_G) = \sum_{i=1}^{N} q_i J_i(V_i) \qquad \text{subject to } V_1 = \cdots = V_N, \tag{19}$$

where $J_i(V_i) = \| V_i - (R + \gamma P V_i) \|^2_{D_i}$ are the local objective functions, $q_i > 0$, $i = 1, \ldots, N$, a priori defined weighting coefficients, and $D_i \in \mathcal{R}^{3K}$invariant probability distributions of each agent's Markov chain.

Building on this foundation, we introduce a novel algorithm for estimating of V^π designed to minimize local objective functions (Equation (19)). This algorithm is an adaptation of a general framework proposed in [13], [44], [64] tailored to address our specific problem. The proposed approach relies on the development of a distributed stochastic gradient method, which integrates Temporal Difference (TD) updates with consensus iterations:

$$V_i(n)'^{(s_i(n))} = V_i(n)^{(s_i(n))} + \alpha q_i \rho_i(n) \delta_i(n) \tag{20}$$

$$V_i(n+1) = \sum_{j \in \mathcal{N}_i} c_{ij}(n) V_j(n)', \tag{21}$$

where α is (typically small) step size, $\delta_i(n) = r_i(n) + \gamma V_i^{(s_i(n)')} - V_i^{(s_i(n))}$ is the TD error, $V_i^{(s)}$ and $V_i'^{(s)}$ mark the estimates of the value function at state s before and after local update (Equation (20)), respectively, $r_i(n)$ is the reward received by node i at step n, $s_i(n)$ is the state of SU_i at time n, $s_i(n)' = s_i(n+1)$, and $\rho_i(n)$ is the importance sampling ratio at step n for agent i. The initial conditions for the recursions are arbitrary.

The $c_{ij}(n)$ parameter denotes the gain at step n which is used by agent i to weight the estimates received from agent j. It is important to note that this communication occurs via a network dedicated to signaling, and not within the licensed spectrum sensed by the SUs. The mentioned parameters are generally random: they take predefined constant values if the consensus communication at step n succeeds (with probability $p_c^{(i)}(n)$), and are set to zero if the communication fails (with probability $1 - p_c^{(i)}(n)$. Also, $c_{ij}(n) \equiv 0$ if $j \notin \mathcal{N}_i$, i.e., if node j is not a neighbor of node i.

As with Equations (8) and (9), the presented algorithm includes two basic steps:

1) Local parameter update (Equation (20)), based on MDP transition and a locally received reward, and
2) In the consensus step (Equation (21)), local communication occurs between neighboring nodes over the dedicated network. This step aims to achieve global parameter estimation in a decentralized manner through distributed agreement among the agents, see Figure 6-2.

As demonstrated in Chapter 4, the algorithm has been proven to converge under several general assumptions. For the specific problem setup considered here, convergence is ensured under the conditions outlined in Assumptions A1, A4, A5, and A6 (formulated in Chapter 4)

The irreducibility condition in Assumption A1.b ensures that the underlying MDPs, under all behavior policies, allow agents to explore all states. The second part guarantees that importance sampling ratios are well-defined. In practical terms, each agent typically focuses on a subset of the overall state space (channels), with high visit probabilities for certain states and lower probabilities for others. This complementary structure among agents can significantly enhance convergence speed.

for *i=1:N* **do**
  initialize $V_i(1)$ and $s_i(1)$
end
for *k=1:episode length* **do**
  for *i=1:N* **do**
    sample and apply an action using policy π_i in state $s_i(n)$
    receive reward $r_i(n)$ and observe state transition to $s_i(n)'$
    update $V_i(n)^{(s_i(n))}$ according to Equation 20
    send updated $V_i(n)'$ to all the nodes j for which $i \in \mathcal{N}_j$ (out-neighbors of node i)
  end
  for *i=1:N* **do**
    apply consensus step on the received $V_j(n)'$, $j \in \mathcal{N}_i$, according to Equation 21
  end
end

Algorithm 1: Distributed consensus-based policy evaluation

Figure 6-2 Algorithm 1 - Distributed consensus-based policy evaluation [43]

By appropriately selecting weighting factors q_i and $c_{ij}(n)$ nodes with higher confidence in accurately sensing primary user (PU) activities on specific channels can be prioritized. Convergence rates can be further enhanced by designing overlapping partitions of the global state space (frequencies) to encourage complementary behaviors among agents. Additionally, the flow of information across the network plays a pivotal role; implementing multi-step consensus among nodes during measurement intervals proves advantageous when inter-node communication rates can be high [144].

Even when agent behaviors do not fully complement each other and states visited by different agents overlap, the "denoising" effect provides another advantage of the consensus-based approach. Consensus-based estimation algorithms generally exhibit strong "denoising" properties, leading to reduced asymptotic variance in the estimates [68], [69]. This variance reduction is a well-known challenge in TD-based algorithms [9], [145] and has been proven beneficial in similar consensus-based schemes, as shown in previous research [13], [64].

6.2.2 Distributed Consensus-based Q-Learning

The scheme discussed in the previous section extends naturally to action-value function Q enabling direct derivation of an optimal policy. The well-known Q-learning algorithm [9], a single-agent approach, is based on Equation (4) and applies TD-based reasoning, similar to the TD learning for state-value functions seen in Equation (20). However, since the goal here is to determine the optimal policy, a key distinction is that, rather than selecting actions based on a predefined policy (as in policy evaluation), actions are chosen through an exploration/exploitation strategy.

To achieve distributed optimal policy search within our CRN JSS model, the algorithm prescribes substitution of local iterations in Equation (20) with the following Q-learning iterations at the local level:

$$Q_i'(n)^{(s_i(n),a_i(n))} = Q_i(n)^{(s_i(n),a_i(n))} + \alpha q_i \left(r_i(n) + \gamma \max_a Q_i(n)^{(s_i(n),a)} - Q_i(k)^{(s_i(n),a_i(n))} \right) \quad (22)$$

$$Q_i(n+1) = \sum_{j \in \mathcal{N}_i} c_{ij}(n) Q_j'^{(n)} \quad (23)$$

In this setup $Q_i(n)$ and $Q_i'^{(n)}$ are matrices representing the i-th node's estimates of the Q-values for all possible state-action pairs, before and after applying the local update (Equation (22)), respectively. $Q_i(n)^{(s,a)}$ denotes the i-th node's estimate of the optimal action value for the state-action pair (s,a) after the consensus update has been applied in the previous step. The parameters α and q_i remain consistent with those defined in the preceding section.

At each time step, an agent performs a local update as per Equation (22), adjusting its estimation of the Q-function for the current state and action pair, $s_i(n)$ and $a_i(n)$. It then receives the Q-values from neighboring agents and performs an iteration consensus (Equation (23)) to update its estimates for all state-action pairs. Initial conditions can be set arbitrarily. However, higher initial values can encourage exploration when using a greedy policy. Additionally, these initial conditions may be reset once the first reward is observed for a specific state-action pair [9].

In standard Q-learning, actions at each step k are chosen using ϵ -greedy strategy [9]. In the multi-agent setting, a modified ϵ-greedy approach is proposed here. To enhance overall performance, each agent, when exploring with probability ϵ, prioritizes a distinct set of channels. This strategy accelerates convergence by leveraging complementary coverage of the state space across agents, akin to the complementary behavior policies employed in policy evaluation. The effectiveness of this approach is demonstrated in the simulations section.

The consensus step in the algorithm retains the same structure as described in Equation (21) (and Equation (9) originally), with the key difference that, in each iteration, agents share their Q-function estimates for all possible state-action pairs (see Figure 6-3). A similar approach has been explored in previous literature [142], accompanied by a comprehensive convergence analysis. However, earlier studies made the limiting assumption that the action selection strategy was fixed beforehand and was independent of the agents' current Q-function estimates.

The distributed Q-learning algorithm proposed here preserves all the advantageous properties discussed in earlier sections.

6.2.3 Convergence Rate and Complexity Analysis

As highlighted in sources such as [146], the non-asymptotic convergence rate —

representing how quickly the "mean path" of the TD or Q-learning algorithm approaches its limit — exhibits an exponential trend. The asymptotic convergence rate can be analyzed through an asymptotic stochastic differential equation, which establishes a direct connection between the asymptotic covariance of the value function estimate and the network's connectivity [64].

In literature related to consensus frameworks, network connectivity is commonly quantified by algebraic connectivity, defined as the second smallest eigenvalue of the graph Laplacian [147]. Greater algebraic connectivity is associated with a faster asymptotic convergence rate and a more pronounced denoising effect in consensus processes (see References [64] [68] for additional details).

The computational complexity of the policy evaluation algorithm, as described in Equations (20) and (21), is $O(K)$, as each step involves updating $3K$ scalar values (one for each state's value function estimate). For the distributed Q-learning algorithm in Equations (22) and (23), the complexity increases to $O(K^2)$, as it tracks the value for each state-action pair, amounting to fewer than $3K \times (K+2)$ computations. Notably, both algorithms are fully scalable concerning network size; their computational complexity remains unaffected by the number of agents, assuming the interconnecting network is sparse (i.e., the network size does not affect the average number of neighbors).

for *i=1:N* **do**
  initialize $Q_i(1)$ and $s_i(1)$
end
for *k=1:episode length* **do**
  for *i=1:N* **do**
    sample and apply ϵ-greedy action $a_i(n)$ in state $s_i(n)$
    receive reward $r_i(n)$ and observe state transition to $s_i(n)'$
    update $Q_i(n)^{(s_i(n),a_i(n))}$ according to Equation 22
    send updated $Q_i'(n)$ to all the nodes j for which $i \in \mathcal{N}_j$
     (out-neighbors of node i)
  end
  for *i=1:N* **do**
    apply consensus step on the received $Q_j'(n)$, $j \in \mathcal{N}_i$, according to
     Equation 23
  end
end

Algorithm 2: Distributed consensus-based Q-learning

Figure 6-3 Algorithm 2 - Distributed consensus-based Q-learning [43]

6.3 Simulations

We explore the characteristics of the proposed consensus-based MARL algorithms for JSS. The analysis considers a scenario with $K = 6$ licensed channels, with channel parameters outlined in Table 2, and sets the parameter $MAX\_ATT = 7$. The setup involves $N = 6$ learning agents (SU_i). For simplicity, the reward ζ for accurately identifying an idle channel state is assigned a value of 1. Additionally, we assume that the duration of all state transitions corresponds to a single time step within the

simulation.

Channel	1	2	3	4	5	6
α	10	5	2	10	5	2
β	2	5	10	2	5	10
PER	0.5	0.5	0.5	0.1	0.1	0.1

Table 2 Channel parameters [43]

In the first simulation, the agents estimate the state-value function in an off-policy setting, without prior knowledge of the model parameters. This estimation is performed using the proposed Algorithm 1, with parameters set to $\alpha = 0.02$ and $\gamma = 0.9$ .The target policy adopted assigns uniform probabilities to different types of actions and equal probabilities for switching to various channels $(i, j = 1, \dots, K, j \neq i)$ (Table 3).

State/Action	SENSE	TRANSMIT	SWITCH_{f_j}
(f_i, IDLE)	1/3	1/3	1/15
(f_i, BUSY)	1/2	0	1/10

Table 3 Target policy parameters [43]

The behavior policies of different agents, denoted as π_j, $j = 1, \dots, N$, are similar to the target policy π, but with a notable distinction: each agent has a specific channel preference, resulting in a significantly higher switching probability for its preferred channel compared to others. In this setup, agent j favors channel f_j with:

$$\pi_j\left(\text{SWITCH}_{f_j}|(f_i,\cdot)\right) = 20\pi_j\left(\text{SWITCH}_{f_k}|(f_i,\cdot)\right), \tag{24}$$

for $i, j, k = 1, \dots, K, j \neq i, k \neq i, j$. When agents are already tuned to their preferred channel $(j = i$ and $\pi_j\left(\text{SWITCH}_{f_j}|(f_i,\cdot)\right) = 0)$ the probabilities for the SENSE and TRANSMIT actions are scaled up. This adjustment ensures that the values of $\pi_j\left(\text{SWITCH}_{f_k}|(f_i,\cdot)\right)$ for other channels remain consistent, as specified in Table 4.

State/Action	SENSE	TRANSMIT	SWITCH_{f_j}	SWITCH_{f_k}
(f_i, IDLE)	1/3	1/3	5/18	1/72
(f_i, BUSY)	1/2	0	5/12	1/48

Table 4 Behavior policies parameters [43]

We begin by evaluating a set of N independent temporal-difference (TD) algorithms, where each algorithm performs only the local TD update step (Equation (20), without including the consensus update (Equation (21)). The resulting state-value function estimates are illustrated in Figure 6-4. The data reveals that each channel is associated with an agent that favors switching to it, leading to faster estimation convergence for that agent relative to others. This effect is most pronounced in states marked as UNKNOWN. Moreover, the estimated values for these unknown states tend

to increase when the corresponding channels are idle and decrease when the channels are occupied. These estimations exhibit significant variability across agents, reflecting notable differences in value function estimates based on each agent's channel preferences.

The performance of Algorithm 1 is then simulated, by the addition of application of the consensus steps (Equation (21)) after the application of the agents' local TD algorithms (Equation (20)). The ring topology is implemented for the communication setup between agents for the purposes of their mutual exchanges of local TD processing information. In this assumed topology, each agent interfaces to the two of its nearest neighbors (i.e., agent 5 interfaces to agent 4 and agent 6). The consensus weights $c_{ij}(n)$ are set to values of 1/3 for the connected agents, and 0 for the ones which are not connected. The weights are presumed to be invariant in time. Figure 6-5 presents the estimations of value functions by agents resulting from application of Algorithm 1. Significant reduction in the variation of estimates across agents compared to the case without consensus is clearly visible.

To provide a more detailed quantitative comparison, we average the value function estimations' mean-square error (relative to the actual values which can be derived by solving the appropriate Bellman expectation equation (Equation (2)) for the given model) across all agents and states. The results, with and without the consensus scheme, are presented in Figure 6-6. The introduction of the consensus mechanism clearly accelerates the convergence of the algorithm. For reference, the performance of a centralized scheme, assuming all-to-all communications, is also included. The results demonstrate that the consensus-based approach achieves outcomes that are nearly equivalent to those of the centralized scheme.

Additionally, the mean variance of value function estimations averaged over all states and across different agents, is also shown in Figure 6-6 for both cases (with and without the consensus scheme). The data reveal that the consensus-based approach reduces estimation variance by approximately two orders of magnitude, highlighting its effectiveness in harmonizing agent estimates.

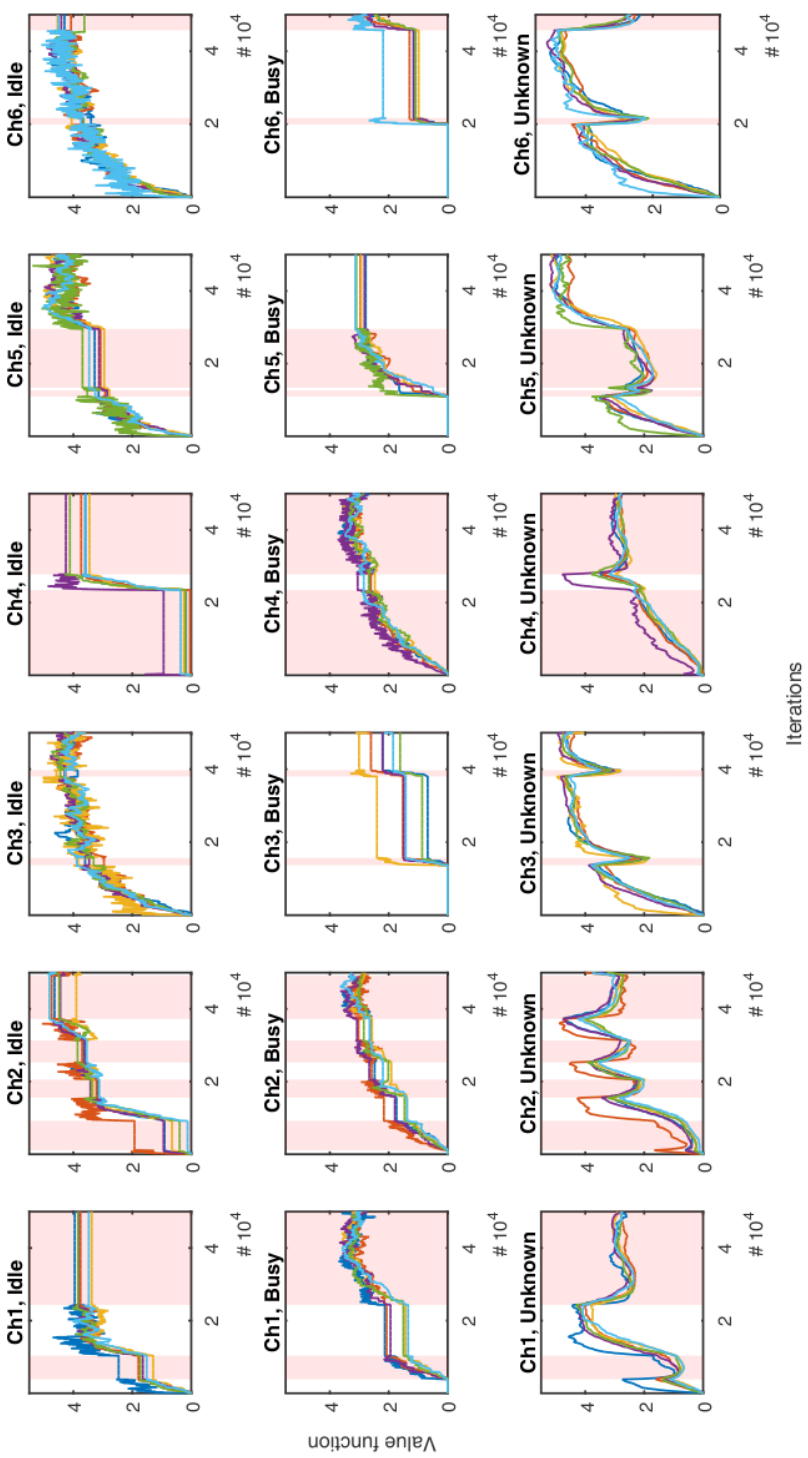


Figure 6-4 Estimations of the value function by all the agents (without consensus) are shown as different lines for all the states, which are shown on different plots. Times while the channels are occupied by primary users are shaded in red. [43]

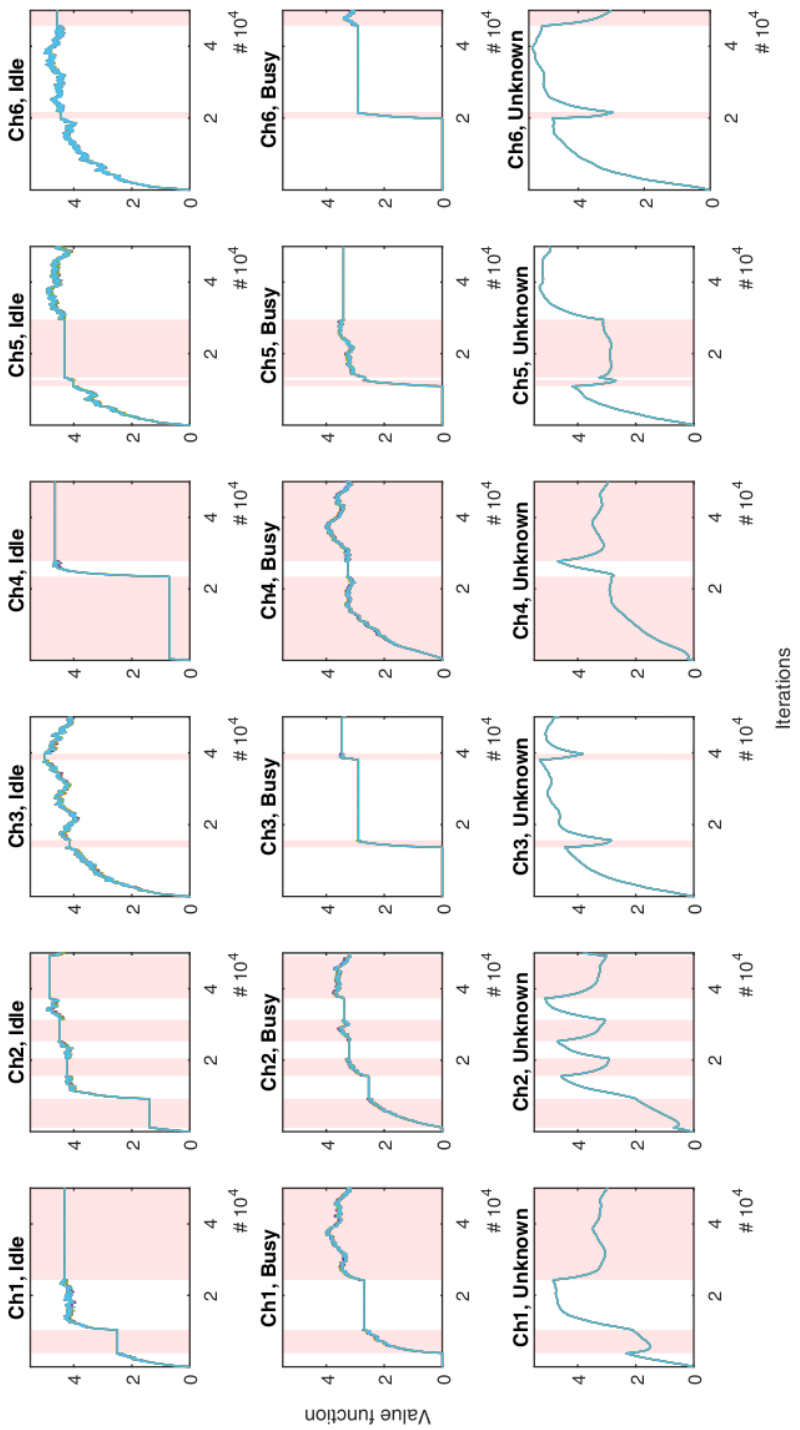


Figure 6-5 Estimations of the value function by all the agents with consensus scheme implemented are shown 88 different lines for all the states, which are shown on different plots. Times while the channels are occupied by primary users are shaded in red [43]

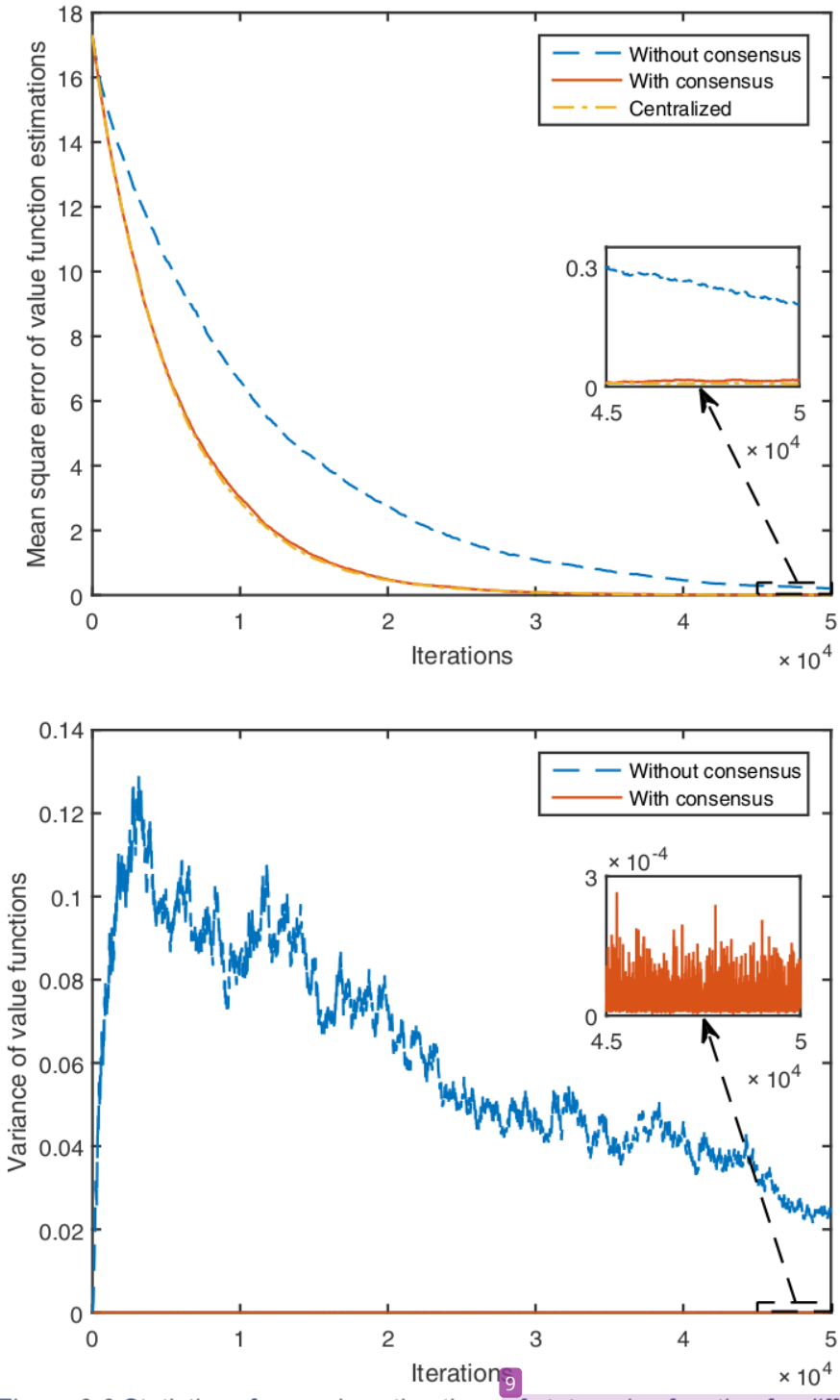


Figure 6-6 Statistics of errors in estimations of state-value function for different algorithms as a function of the number of iterations [43]

Consensus schemes are effective at providing reliable estimates even to agents without direct observation access to certain variables of interest. In a setup where local behavior policies are arranged so that each channel is accessible by three agents that can switch to it, while the other three agents never access it, there are three active estimates for each state-value function. For example, one state illustrated in Figure 6-7 demonstrates that the remaining three estimates, initialized to zero, remain unchanged. The results confirm that the consensus scheme enables all agents to obtain meaningful estimates of state-value functions under these conditions. Notably, in this setup, the aspect of Assumption A1 concerning importance sampling ratios (often referred to as the coverage assumption) is evidently unmet. However, the consensus approach allows this coverage to be reached at the network level, facilitating effective and feasible implementation of Algorithm 1.

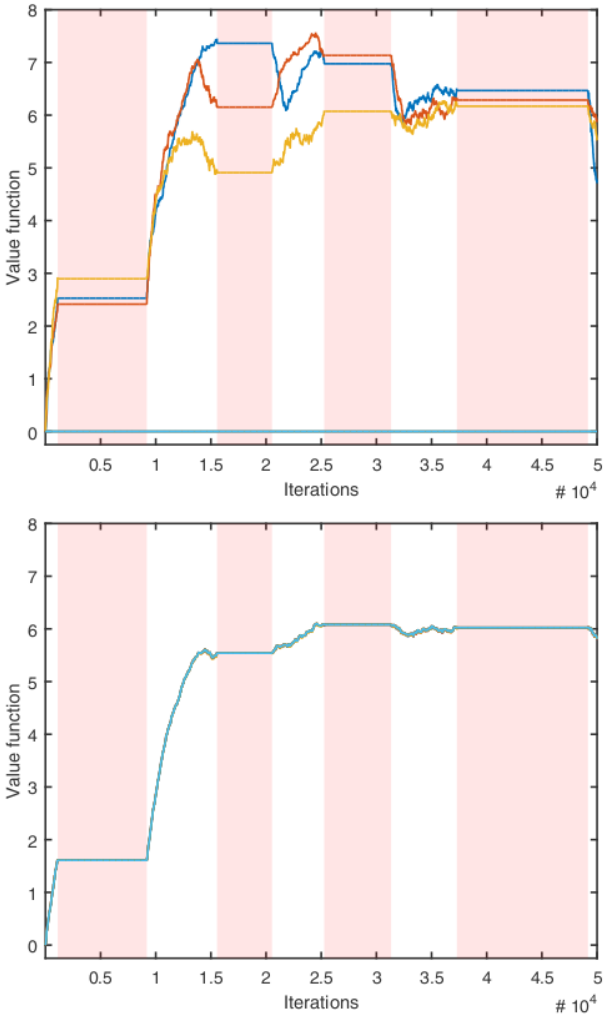


Figure 6-7 Estimations of the value function performed by all agents both without consensus (upper figure) and with consensus (lower figure), shown with different lines, for a state which can only be visited by three agents. Times while the channel in question is occupied by primary users are shaded in red. [43]

In the following experiment, Algorithm 2 is used to estimate the action-value function with parameters $\alpha = 0.2$, $\gamma = 0.9$ and $p_D = 0.9$. Each agent applies an individual ϵ-greedy behavior policy π_i, $i = 1, \dots, N$, with $\epsilon = 0.5$. The setup mirrors the first experiment, where each agent has a preferred channel for exploratory actions, switching to this channel with a probability 20 times higher than for other channels.

Initially, we examine the case without consensus, where agents perform only local processing (Equation (22)) without inter-agent communication, as shown in Figure 6-8. To keep the figure clear, we display Q-value estimates for actions involving a single channel. As expected, the agents' action-value functions show considerable variation due to their distinct exploratory policies.

In the scenario with the consensus communication scheme enabled, depicted in Figure 6-9, the consensus mechanism enables substantial alignment among the agents' estimates. This is evident as the partially transparent lines, each representing a different agent's action-value estimate, mostly overlap (resulting in $N = 6$ overlapping lines in each plot). Additionally, intervals corresponding to the availability of destination channels during switching actions are illustrated. Here, the action-value estimates adjust based on channel availability, increasing when the destination channel is idle and decreasing when it is busy, as seen in parts of $Sw.\ to\ Dest.\ |\ Ch1,\ Idle$ plots when $Ch1$ is idle and in parts of $Sw.\ to\ Dest.\ |\ Ch1,\ Busy$ plots when $Ch1$ is busy.

In Figure 6-8 and Figure 6-9 the estimates of the action-value function are presented for all the possible actions from the three possible states, which are all associated to one of the used channels – titles of subplots name the Action|State pair there presented. Estimates of different agents correspond to lines of different color. Times while the channel in question is occupied by primary users are shaded in red. Times during which the destination channel of the switching action are Intervals when the channels corresponding to destinations of switching actions are occupied by primary users are shaded in gray.

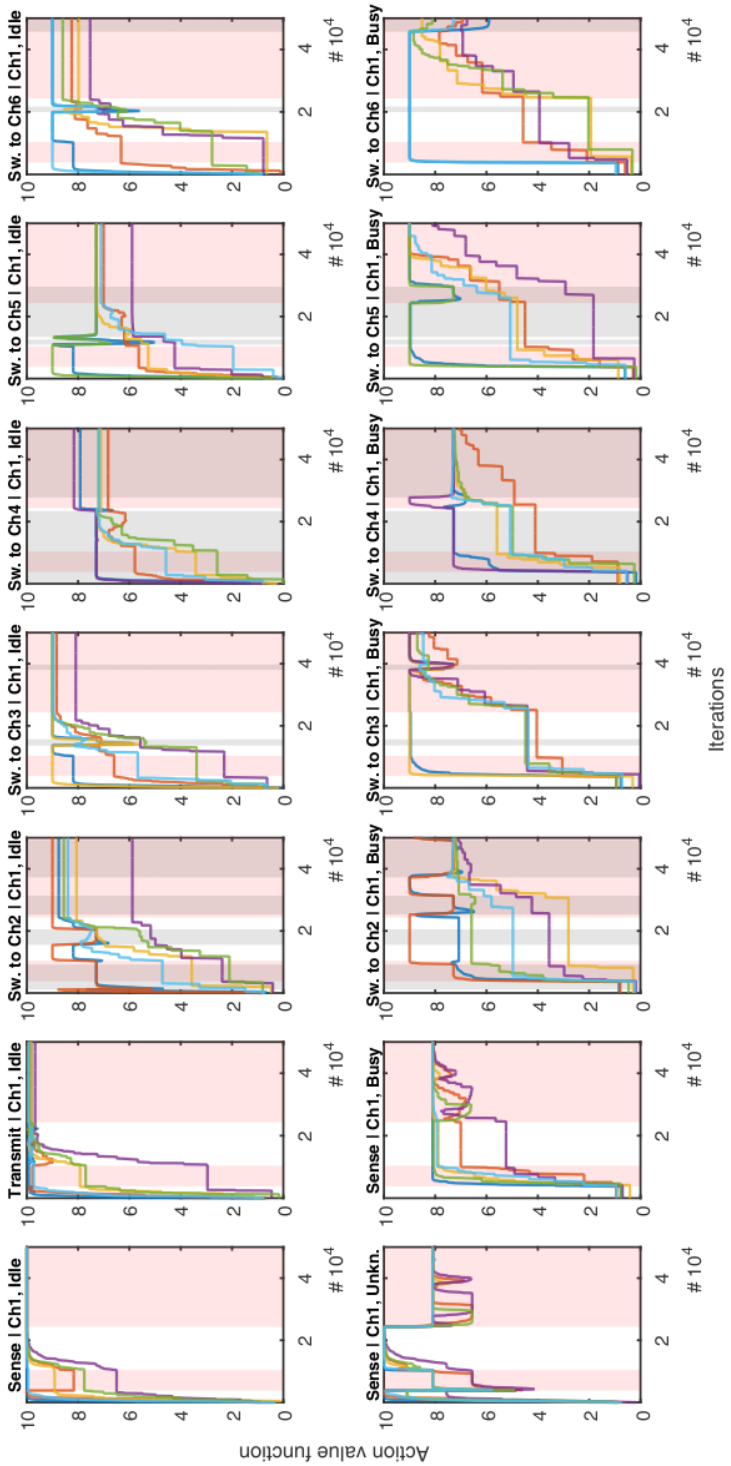


Figure 6-8 Estimations of action-value function for one channel by all the agents (without consensus) [43]

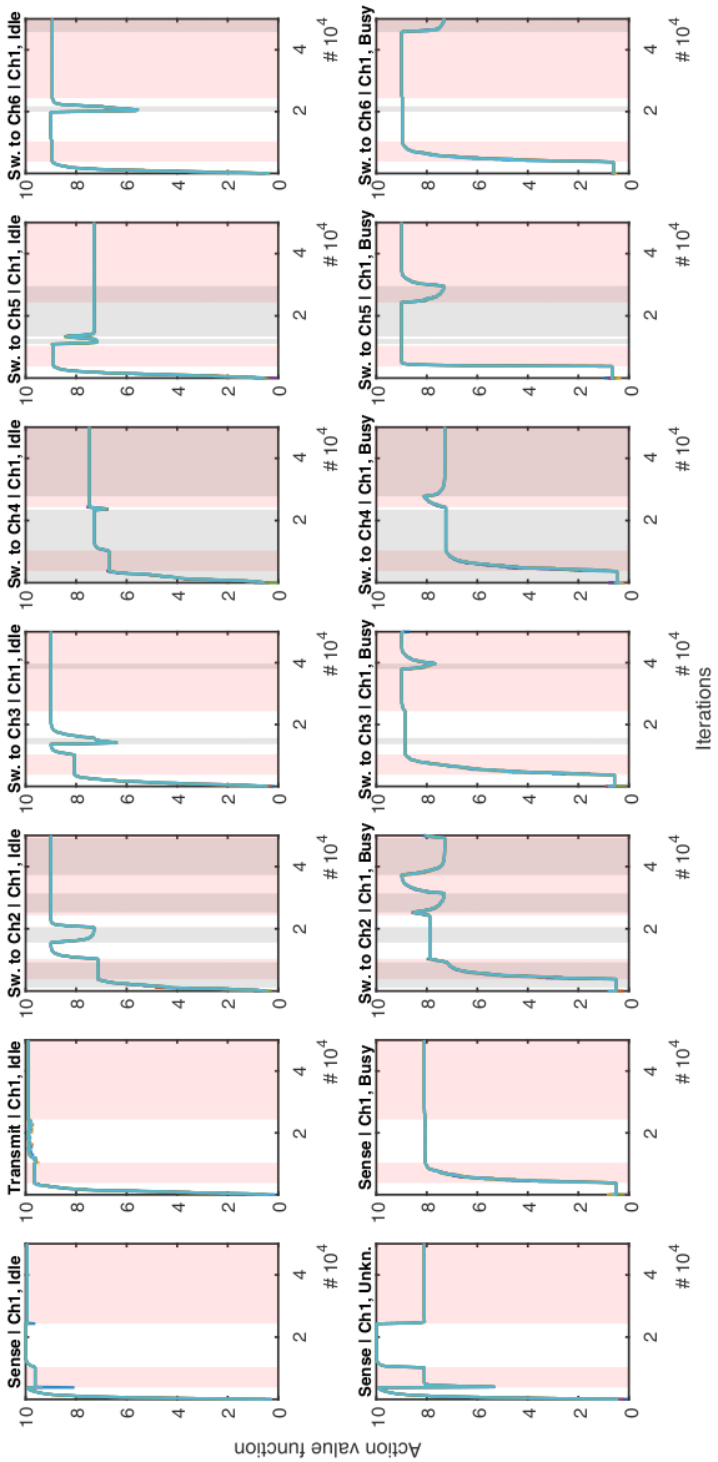


Figure 6-9 Estimations of action-value function for one channel by all the agents (with consensus)

To evaluate the algorithm's performance in the context of the presented JSS model, the number of successful and failed transmissions is measured (failures due to interference with primary users) for various algorithms. The results of these measurements over entire networks are presented in Figure 6-10. The proposed scheme is compared to a baseline, which represents a recently introduced decentralized cooperative approach that aggregates local rewards without consensus (adapted from the algorithm in [31] for the proposed system model and similar to the information sharing scheme from [2], labeled as "Without consensus + Coop").

The results demonstrate that incorporating the proposed consensus mechanism increases throughput by approximately 50% within the initial 10,000 iterations of the algorithm. As expected, the centralized scheme achieves slightly superior performance. However, the baseline cooperative scheme performs worse than the consensus-based approach proposed in the thesis due to high variance in Q-value estimates inherent to this scheme.

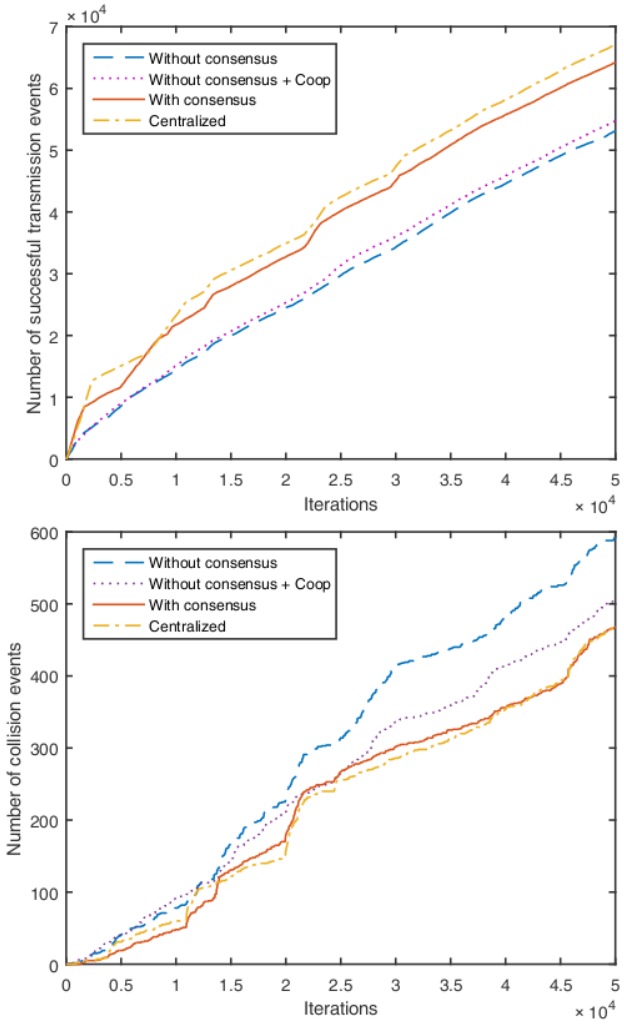


Figure 6-10 Total count of transmission events within the network, both successful and failed, for different algorithms as a function of the number of iterations [43]

The breakdown of successful and failed transmissions for the consensus-based algorithm across individual channels is provided in Table 5. It is evident that channel 6, characterized by low primary user activity and a low packet error rate (PER), emerges as the preferred channel for transmission actions.

Channel	1	2	3	4	5	6
successful transmissions	1440	2263	2885	115	11,739	45,678
failed transmissions	75	136	8	74	94	85

Table 5 Contributions through participating channels within the network to the total number of successful and failed transmissions [43]

It is important to highlight that the demonstrated advantages of the presented consensus communication scheme — namely improved coordination, reduced disagreement among agents, and accelerated convergence of the resulting algorithms — extend beyond the specific parametrization and system model adopted in this study. The chosen model, inspired by [2], was selected to clearly illustrate the features of the presented scheme and the associated estimates of the action-value function. However, various alternatives can be integrated into the presented consensus-based framework.

For instance, reducing ζ, the sensing reward for identifying an idle channel, would relatively increase the rewards for transmitting actions, potentially leading to higher values in Table 5. Additionally, sensing errors could be more effectively addressed by introducing penalty terms for transmitting on busy channels. The assumed primary user (PU) activity model could also be refined, yielding improved local processing results. Experiments using alternative models derived from historical PU activity statistics [148] show that the comparative results remain consistent with those depicted in Figure 6-10.

Furthermore, more advanced spectrum sensing techniques, such as machine learning-based methods [149], could replace the energy detection scheme used in this study. However, even with improved local sensing algorithms, the performance margin between consensus-based and non-consensus approaches observed in Figure 6-10 would persist. This is primarily due to the high variance in action-value estimates characteristic of algorithms without consensus.

7. SPECTRUM SENSING BY MODULATION CLASSIFICATION USING DEEP NEURAL NETWORKS

The growing global demand for seamless information exchange among mobile users has driven rapid advancements in wireless communication. The deployment 5G mobile communication is well underway, while research and development efforts for sixth-generation (6G) technology have already begun.

5G technology not only enhances human-centered communication by delivering a significantly improved user experience but also fosters the emergence of many applications, from telephony and SMS to multimedia [150].

7.1 Challenges in Spectrum Management and Utilization

The growth of the breadth of services and their adoption has led to a dramatic rise in the number of mobile phone users. The electromagnetic spectrum, a critical and limited resource for wireless communication, is facing growing scarcity. The rapid growth of wireless services has led to a significant increase in demand for spectrum bandwidth [151].

According to the International Telecommunication Union (ITU), the available spectrum ranges from 9 kHz to 275 GHz. In its 2006 forecast, the ITU estimated that global spectrum demand would range between 1,280 MHz and 1,720 MHz by 2020. The exponential growth in user data traffic, these predictions have fallen short. Currently, available spectrum resources meet only about half of the actual demand [152]. This shortage has become a critical challenge, potentially hindering the sustainable growth of wireless communication technologies, if not effectively addressed.

Spectrum resources worldwide are regulated by governmental agencies. These agencies generally utilize a "static allocation" approach, dividing the spectrum into distinct, non-overlapping frequency bands designated for specific users. These designated users, known as "authorized users," operate exclusively within their assigned frequency bands.

Despite efforts to allocate spectrum rationally, the utilization rates of assigned frequency bands vary significantly across times and regions. A study by the FCC revealed that the utilization of licensed bands, particularly in the highly demanded range of hundreds of megahertz to 3 GHz, fluctuates between 15% and 85%. In some areas, utilization is as low as 5%. This inefficiency, combined with the uneven distribution of spectrum usage, highlights the pressing issue of underutilized resources.

Most spectrum resources below 1 GHz have already been allocated, leaving limited bandwidth available for new wireless services. Addressing this challenge requires transitioning from static spectrum allocation to dynamic spectrum management and sharing mechanisms. These approaches aim to optimize spectrum utilization across different times and regions, ensuring more efficient use of this finite resource and supporting the continued expansion of wireless communication technologies.

7.2 Strategic Role of Spectrum Sensing

Spectrum sensing (SS) has extensive applications across multiple domains, with significant research efforts dedicated to advancing this technology. The development of SS technology is strategically vital, as it is poised to become a foundational element in numerous fields, including communications, radar, UAVs, and smart transportation. By enabling efficient spectrum utilization, SS not only addresses spectrum scarcity, but also supports innovation in emerging technologies, laying the groundwork for a more connected and intelligent future.

In communication systems, SS can manage and allocate spectrum resources dynamically, improving spectrum utilization. By identifying unused primary user bands, it enables better resource allocation and supports dynamic spectrum sharing.

In radar systems, SS monitors and analyzes radar spectrum use, facilitating optimal spectrum management and countering jamming attempts [153].

For unmanned aerial vehicles (UAVs), SS provides crucial real-time spectrum information to select optimal communication bands and avoid interference [154].

In intelligent transportation networks, SS monitors spectrum usage by various wireless devices, enabling dynamic spectrum management. This ensures communication reliability and quality for diverse applications, from traffic management to autonomous vehicle coordination [155].

7.3 Integration of Deep Learning into Spectrum Sensing

In recent years, artificial intelligence, particularly deep learning, has transformed spectrum-sensing research. With its capabilities in nonlinear modeling and adaptive learning, deep learning has greatly improved the performance and efficiency of spectrum-sensing techniques, facilitating more effective utilization of spectrum resources [156]. Several advanced deep learning models have been applied to spectrum sensing, most prominent being:

- Convolutional Neural Networks (CNNs): These have demonstrated exceptional capabilities in feature extraction for spectrum analysis.
- Long Short-Term Memory Networks (LSTMs): Leveraging their ability to process sequential data, LSTMs have been effective in dynamic spectrum environments [157] [158].

Although spectrum sensing utilizing classical DL networks has shown promising outcomes, it is not without its limitations. The majority of existing research primarily emphasizes the integration of deep learning with non-cooperative spectrum sensing, which introduces several significant challenges:

- Limited Availability of Labeled Data: Individual devices rely solely on their local sensing results, restricting data sharing and collaboration. This often leads to a scarcity of large-scale labeled datasets required to train robust deep learning models.

- Resource Constraints: Non-cooperative devices typically operate independently, and their computational and energy capacities may fall short of the requirements for executing deep learning models effectively.
- Exploring Cooperative Spectrum Sensing with Deep Learning: To address these limitations, researchers have increasingly focused on applying DL to cooperative spectrum sensing (CSS). Cooperative sensing combines data from multiple devices, resulting in a more comprehensive and diverse dataset for deep learning training. This approach provides several advantages:
- Real-Time Adaptability: In real-time environmental updates, enabling deep learning models to adjust dynamically to changing conditions.
- Improved Data Availability: By integrating multi-source observational data, CSS enhances DL higher accuracy.

These findings indicate that cooperative spectrum sensing, powered by deep learning, can address the shortcomings of non-cooperative methods. By fostering collaboration between devices and leveraging real-time data, CSS offers a promising path forward for efficient spectrum utilization in dynamic and resource-constrained environments.

Recent research on deep learning for spectrum sensing primarily emphasizes the development and utilization of advanced neural network architectures. This section explores the application of widely used models, including convolutional neural networks (CNNs), residual networks (ResNets), long short-term memory networks (LSTMs), and other deep learning frameworks.

7.4 Application of Deep Learning Frameworks in Spectrum Sensing

Convolutional Neural Networks (CNNs) are a type of artificial neural network (ANN) widely recognized as one of the most effective deep learning models [159]. CNN are preferred choice for tasks such as object detection, image recognition, and speech recognition.

A typical CNN architecture consists of three primary components:

- Convolutional Layers, which extract features by applying convolutional filters. Each layer identifies patterns such as edges or textures, building progressively more complex representations of the input data.
- Pooling Layers, which downsample the output from convolutional layers, reducing the spatial dimensions of the data. This process preserves essential features while decreasing computational complexity and helping to prevent overfitting. However, it also results in some loss of information.
- Connected Layers, integrating all the features extracted by previous layers, producing a global feature representation. They are typically used at the final stage of the network to perform tasks such as recognition or classification based on the extracted features.

In CNNs, each layer progressively transforms the input data into high-level abstractions. The architecture showcases the ability of CNNs to identify and classify features effectively, making them a powerful tool for spectrum sensing.

In signal processing, covariance matrices are fundamental tools. By reflecting the second-order statistical properties of signals, covariance matrices serve as a key statistical descriptor. Recent studies have demonstrated their effectiveness in spectrum sensing, particularly in scenarios with low signal-to-noise ratios (SNRs) and noise uncertainties, often outperforming traditional methods. Furthermore, the robust feature extraction capabilities of convolutional neural networks (CNNs) have made covariance matrix-based spectrum sensing methods a promising area of research.

The spectral correlation function (SCF), which is derived from a signal's autocorrelation function, is commonly used to characterize the spectral properties of periodic statistical signals. Recent advancements have combined SCF with CNNs to improve spectrum sensing performance.

In Spectrum Sensing, Residual neural networks (ResNets) address the vanishing gradient problem in deep networks, enabling the development of deeper architectures. ResNets excel in extracting features from two-dimensional spectral data, making them highly effective for spectrum sensing.

Long Short-Term Memory (LSTM) Networks, a type of recurrent neural network (RNN), model long-term dependencies in data. Their ability to process complex, noisy spectral data makes them particularly suitable for spectrum sensing tasks. LSTM networks leverage their internal memory to predict and adapt to dynamic spectrum environments, ensuring high accuracy in detecting spectrum states [160].

Novel methods of spectrum sensing leverage LSTM networks to capture temporal correlations in spectrum data. By linking prior sensing events to current information, they effectively model temporal dependencies in cognitive radio environments. Similarly, in [161], a spectrum-sensing framework employing LSTM networks was proposed to handle temporal signal dependencies. This approach demonstrated superior performance under low SNR conditions compared to alternative methods. However, it also required longer training and execution times. The study highlighted LSTM's potential for spectrum sensing and recommended future efforts to reduce the computational overhead of such networks.

LSTM networks have also been integrated with other algorithms to improve spectrum prediction, a critical aspect of spectrum sensing. Ling Yu et al. [162] highlighted the ability of LSTM networks to manage dynamic temporal dependencies, making them a vital component in deep learning-based prediction methods. These hybrid approaches have expanded the potential applications of LSTM in spectrum sensing.

7.5 Performance Comparison: Traditional vs. Deep Learning Approaches

Traditional single-node spectrum sensing methods and matched filter detection, are constrained by certain environmental factors. For example, energy detection struggles in low SNR settings, while cyclostationary methods require partial information about PU signals and suffer from complexity.

In contrast, deep-learning-based methods demonstrate broader applicability and superior performance, particularly at low SNRs. Detection probabilities using deep

learning models range from 85% to 90% at -20 dB and nearly 100% at 0 dB, significantly outperforming traditional approaches. However, these methods demand substantial computational resources and large-scale labeled datasets, which remain a challenge for real-world implementation.

7.6 Challenges and Research Perspectives

Despite its potential, deep-learning-based CSS faces several challenges that require further exploration:

- Enhancing Low-SNR Performance: While some CNN-based methods show promise, performance at extremely low SNRs (-13 dB to -20 dB) still declines sharply. Future research should focus on data augmentation and joint reinforcement learning to improve robustness and detection accuracy.

- Strengthening Security Against Malicious Attacks: Techniques like adversarial training, hybrid models, and autoencoders have been employed to defend against attacks like primary user emulation and SSDF. However, adapting these methods to diverse real-world threats remains an open area for investigation.

- Spectrum Sensing with Limited Data: Current approaches heavily rely on large datasets. Innovations such as transfer learning, semi-supervised learning, and advanced data augmentation techniques (e.g., DCGANs) could make deep-learning models viable in data-scarce scenarios.

- Dynamic Threshold Adjustments: Adaptive and intelligent threshold-setting techniques are critical for addressing dynamic environments. Future work must address these practical challenges.

- Expanding Autoencoder Applications: Autoencoders, with their capabilities in feature extraction and noise reduction, hold promise for spectrum sensing. Studies integrating autoencoders with LSTM and CNN models have shown effectiveness in mitigating malicious node impacts, suggesting further research in this area.

As a cornerstone of cognitive radio, spectrum sensing plays a vital role in optimizing spectrum utilization. Traditional methods, while foundational, are increasingly inadequate for complex sensing environments. By integrating advanced deep learning and DRL techniques, researchers have demonstrated significant improvements in detection accuracy, adaptability, and robustness.

However, challenges such as computational complexity, data dependency, and real-time application barriers remain. Addressing these will require continued innovation and collaboration among researchers. Future efforts should focus on enhancing performance in low-SNR environments, mitigating security threats, and improving model efficiency for real-world deployment. With these advancements, deep-learning-based CSS can unlock new opportunities in communication systems, driving more efficient and secure spectrum utilization.

7.7 Deep Learning-Based Modulation Classification Use Case

The thesis proposes a solution to the well-known challenge of modulation classification in radio signals using DNN models. Modulation classification is an ideal problem playing to the strengths of deep learning (DL) architectures, which offer substantial advantages over traditional machine learning techniques bypassing manual feature extraction. The implemented simulated models have been trained using the publicly available RadioML2016.10a time series dataset [37] and build on previous work ([38], [40]) to provide evaluation of various DL models, with a focus on enhancing performance through ensemble averaging.

The task at hand is to classify the modulation type from 11 possible categories using complex (real and imaginary parts) time-domain signals, each consisting of 128 samples. The dataset contains signals at 20 different signal-to-noise ratios (SNRs). The 2x128 input vectors are fed into DL, which are trained end-to-end. We will utilize the TensorFlow framework and Keras library on Google Colab to design and train the models. The performance of the following NNs is evaluated:

- Baseline Convolutional Model (CNN): This model includes 2 layers with 64 1x3 filters and 16 2x3 filters, a fully connected layer with 128 units and a softmax output layer. Dropout regularization with a 0.5 probability is applied between layers to prevent overfitting.
- Deeper Convolutional Model (CNN_deep): This model incorporates 4 convolutional layers with 256 1x3 filters, 256 2x3 filters, 80 1x3 filters, and 80 1x3 filters, followed by a fully connected layer with 128 units and a softmax output layer. Dropout regularization is applied similarly to the previous model.
- Deeper Convolutional Model with LSTM Layer (CNN_deep_LSTM): This architecture includes four convolutional layers with 128 1x3 filters, 64 2x3 filters, 32 1x3 filters, and 16 2x3 filters. It is followed by an LSTM layer with 64 units and a fully connected layer with 128 units before the softmax output layer. Dropout regularization is also implemented.
- Deeper Convolutional Model with LSTM Layer and Ensemble Averaging (CNN_deep_LSTM_ens_cons): This ensemble model comprises three CNN_deep_LSTM networks, each trained on a different augmented version of the original dataset. The augmentation is performed using time-domain techniques like window cropping and flipping [39]. The final output is derived by averaging the softmax results from the individual models. This ensemble method can be efficiently scaled with additional models, using an ad-hoc consensus algorithm similar to the approach in [43].

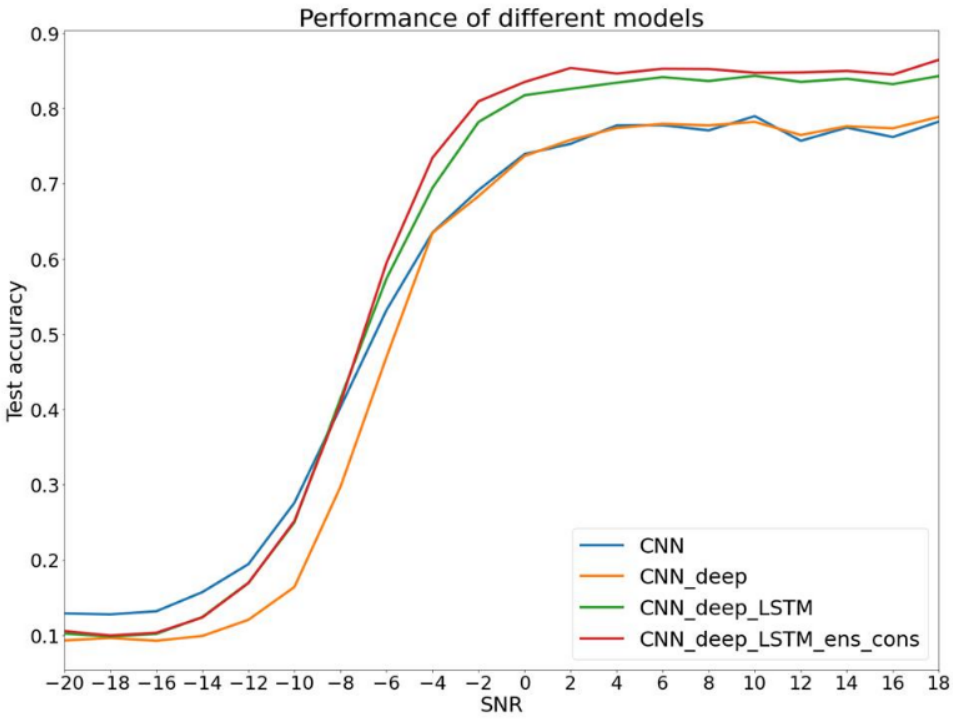


Figure 7-1 Different deep learning models for different SNRs [163]

The results are presented in Figure 7-1, where we display the top-1 classification accuracy on test data (30% of the original dataset) for various models across different SNR of the modulated signals. The results indicate that integrating an LSTM layer significantly improves classification performance providing an additional boost in accuracy. While we also tested a network incorporating residual layers, it failed to deliver comparable outcomes, primarily due to challenges in optimizing its parameters. To provide a clearer understanding of the performance, we present the confusion matrix for the ensemble model's classification results at the highest SNR in Figure 7-2. The matrix reveals strong results, although there is room for improvement. For instance, QAM16 and QAM64 signals are frequently misclassified as one another, and WBFM signals are often confused with AM-DSB.

We demonstrate the potential of deep learning approaches in cognitive radio applications. The results clearly highlight the advantages of DL models, as they outperform manually designed features and traditional machine learning models based on advanced signal processing techniques [38]. Consequently, DL methods now stand out as a leading tool for radio signal classification.

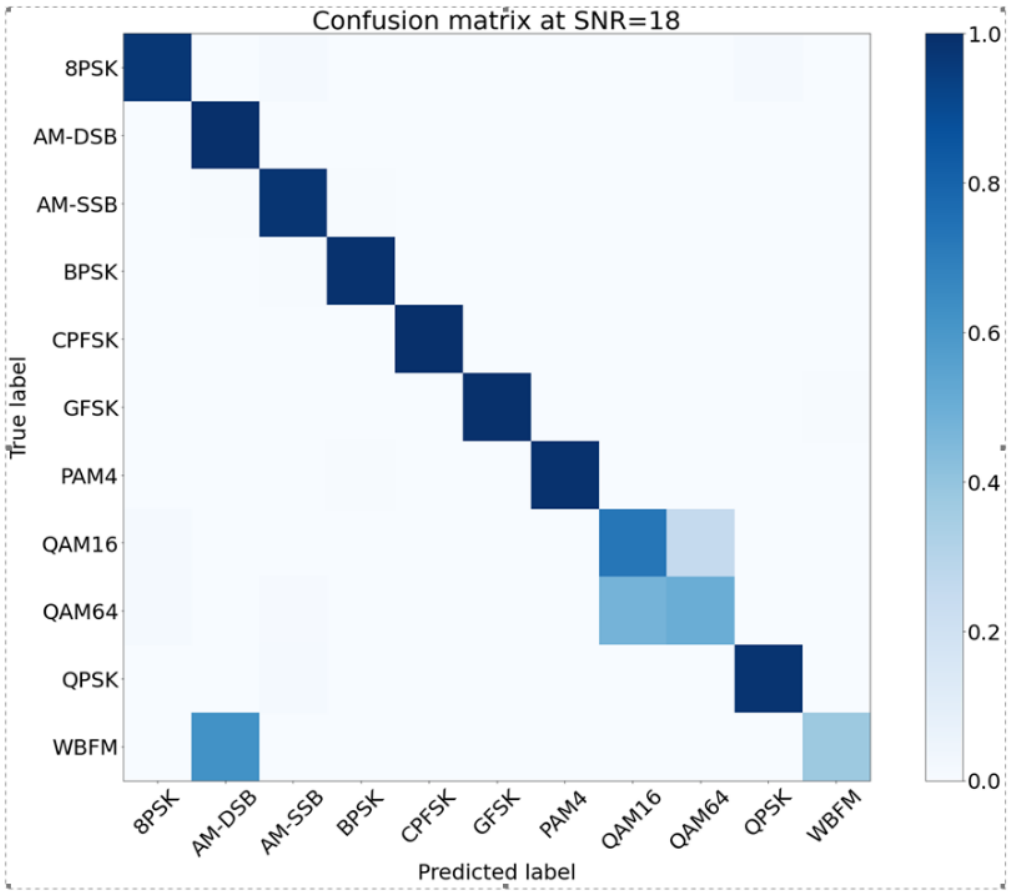


Figure 7-2 Performance of CNN deep LSTM [163]

The application DL techniques across various mobile networking services and architectures. In most of these areas, DL has been shown to outperform traditional methods, although some limitations that hinder its broader applicability have also been identified. Through simulations, we demonstrated that the combination of a deep CNN with an LSTM layer and ensemble averaging improves performance in modulation classification, compared to shallower neural network architectures.

Looking ahead, we foresee several promising directions for DL in the future. One key area is the creation of large-scale, high-quality datasets necessary for training next-generation networks. As networks continue to generate vast amounts of unlabeled data, advancements in unsupervised learning techniques for data labeling and classification are expected to gain momentum. Additionally, further progress in DRL for network control and optimization is anticipated, potentially transforming how networks are managed and operated.

8. CONCLUSION

This dissertation focuses on the development of a novel algorithm for distributed joint spectrum sensing and channel selection in cognitive radio networks. The approach relies on a consensus mechanism among secondary users, which is integrated into a multi-agent reinforcement learning framework. The consensus strategy is facilitated through a directed, time-varying communication network characterized by low bandwidth and throughput, enabling decentralized and distributed cooperation among agents. This approach is motivated by typical cognitive radio network deployment scenarios, where decentralized setups and distributed operations are critical.

The proposed framework equips agents in an unknown environment with actionable insights regarding the network state, allowing the collective computation of an optimal joint strategy for spectrum sensing and channel selection. This is achieved even though individual agents lack sufficient information to do so independently. The algorithm demonstrates scalability and resilience to agent or link failures, ensuring robust operation.

Key algorithmic features, including its denoising capability, ability to coordinate actions, and enhanced convergence rate due to consensus, are thoroughly analyzed. Additionally, the dissertation explores the application of deep learning (DL) techniques to address challenges in spectrum sensing and channel selection in the next generation of radio networks.

To achieve the research objectives and validate the hypotheses, the study employs a theoretical analysis grounded in a comprehensive review of international scientific literature. Insights from previous studies and contributions of experts in this field are integrated to inform the development and evaluation of the proposed solutions. The dissertation combines scientific and theoretical perspectives with original contributions, utilizing various methods such as historical analysis, systematic observation, content analysis, synthesis, proof techniques, and experimental methodologies.

Using the historical method, insights were gathered from the research of other authors who have explored topics related to this study. The data primarily originates from prominent scientific works and studies.

The methods of complex observation and content analysis were employed to process findings from these prior studies. These findings were instrumental in defining the necessary objectives for the newly proposed multi-agent system schemes and associated algorithms, as well as evaluating the effectiveness of the deep learning approach in classification tasks.

By examining the methodologies and outcomes of existing research, opportunities to enhance and apply improved frameworks were identified. This includes the development of novel algorithms for secondary user behavior in cognitive networks, incorporating cooperation based on consensus principles. Furthermore, understanding the behavior of primary users enabled the design of deep learning models for agents, facilitating accurate detection of frequency resource utilization by primary users. This approach integrates the methods of analysis and synthesis to

achieve its objectives.

To assess the validity and applicability of the proposed schemes, a theoretical analysis was conducted using proof methods to establish the existence of desired properties. These properties were confirmed under realistic implementation conditions. Additionally, the effectiveness and efficiency of the proposed frameworks were evaluated through behavioral simulations, which substantiated the initial hypotheses of the dissertation, representing an application of the experimental method. Through extensive simulation and performance evaluation, the proposed framework demonstrates significant improvements in terms of spectrum sensing and channel selection efficiency, and overall cognitive network performance. The findings contribute to the development of more robust and scalable solutions for cognitive radio networks, with applications in next-generation wireless communication systems.

The thesis addresses policy evaluation, as, well as policy optimization tasks. Regarding the policy evaluation, a consensus mechanism is combined with the temporal difference (TD) learning algorithm in an off-policy setting where a single target policy is evaluated while individual agents operate with distinct behavior policies suited to their characteristics. In policy optimization, the consensus approach is applied to action-value function learning, with agents locally implementing the Q-learning algorithm. Each agent explores its preferred set of channels, promoting complementary exploration across the network and accelerating convergence. Notably, even when individual agents lack the capability to independently evaluate or learn the optimal policy due to limited resources or information, the distributed consensus-based approach enables the network as a whole to achieve optimal performance.

Agents exchange their local estimates of value functions—either state-value or action-value—exclusively with their immediate neighbors through low-bandwidth communication links. This fully decentralized framework ensures that each agent depends only on its local observations and the information received from nearby nodes, without the need for centralized coordination. Despite this decentralized structure, the network achieves convergence to a consistent estimate of the target value function after a sufficient number of iterations. In the case of Q-learning, the agents collectively approximate the optimal policy. Moreover, the consensus process inherently provides a denoising effect by aggregating and smoothing local estimates through convexification at each iteration. This mechanism enhances the accuracy and robustness of the learning process.

The proposed algorithm offers several notable benefits. It is scalable, capable of managing large networks, and resilient to failures of nodes or communication links. Its convergence properties, along with the underlying assumptions and limitations, are thoroughly analyzed and discussed. Simulation results demonstrate the advantages of this consensus-driven MARL framework, showing that even with sparse neighbor-based communication graphs, the algorithm achieves performance close to the global optimum. Furthermore, numerical comparisons reveal the superiority of this approach, which relies on exchanging local value function estimates, over alternative cooperative reinforcement learning methods ([2], [31]) that focus on sharing local rewards. The latter often exhibit higher variance in value function estimates, reducing their overall efficiency.

This framework can be understood as serving multiple purposes:

- Coordination Tool: Facilitates the management of actions across multiple nodes or agents, each responsible for specific regions of the state space, while working collectively towards a shared objective.
- Parallelization Tool: Speeds up convergence, making it particularly useful for tackling high-dimensional problems, such as those involving a large number of frequency channels.
- Denoising Tool: Utilizes the averaging of noise realizations across different agents, which is especially beneficial when certain agents experience high error rates in spectrum sensing or elevated Packet Error Rates (PER) on specific channels (e.g., due to fading or shadowing effects). In such cases, decisions from less reliable agents are typically corrected by the majority of nodes operating under more favorable conditions.

As part of the thesis, simulations of deep learning (DL) models were conducted to enable modulation classification or recognition, aimed at implementation within CR network agents. This capability allows secondary users to identify known primary users of the spectrum resource. Various deep neural network architectures were explored and evaluated, comparing their performance to demonstrate the potential of DL methods for learning temporal patterns directly from raw signal data. Using a "blind" temporal learning approach, where raw data samples are input into the network, the models were trained on publicly available datasets, and their results were benchmarked against existing data. The findings indicate that integrating the proposed consensus algorithm with agents running DL models can significantly enhance the accuracy of modulation classification.

Future research will aim to extend these efforts by applying distributed actor-critic schemes to cognitive radio network (CRN) challenges, particularly in environments with substantial inter-secondary-user (SU) interference. Promising directions include exploring TD(λ) methods for adaptive consensus techniques. Additionally, plans include incorporating function approximation for both state-value and action-value functions to address problems involving large state or state-action spaces effectively.

LITERATURE

[1] I. F. Akyildiz, W. Y. Lee, M. C. Vuran, and S. Mohanty, “NeXt generation/dynamic spectrum access/cognitive radio wireless networks: A survey,” *Comput. Networks*, vol. 50, no. 13, pp. 2127–2159, 2006, doi: 10.1016/j.comnet.2006.05.001.

[2] M. Di Felice, L. Bedogni, and L. Bononi, “Reinforcement Learning-Based Spectrum Management for Cognitive Radio Networks: A Literature Review and Case Study,” in *Handbook of Cognitive Radio*, vol. 3–3, W. Zhang, Ed., Singapore: Springer Singapore, 2019, pp. 1849–1886. doi: 10.1007/978-981-10-1394-2_58.

[3] M. Beko, “Efficient Beamforming in Cognitive Radio Multicast Transmission,” *IEEE Trans. Wirel. Commun.*, vol. 11, no. 11, pp. 4108–4117, Nov. 2012, doi: 10.1109/TWC.2012.092712.120201.

[4] M. Beko, “Efficient convex optimization for beamforming in cognitive radio multicast transmission,” in *2012 IEEE International Conference on Communications (ICC)*, IEEE, Jun. 2012, pp. 1895–1899. doi: 10.1109/ICC.2012.6364285.

[5] T. Yucek and H. Arslan, “A survey of spectrum sensing algorithms for cognitive radio applications,” *IEEE Commun. Surv. Tutorials*, vol. 11, no. 1, pp. 116–130, 2009, doi: 10.1109/SURV.2009.090109.

[6] B. F. Lo and I. F. Akyildiz, “Reinforcement learning-based cooperative sensing in cognitive radio ad hoc networks,” in *21st Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC*, Istanbul, 2010, pp. 2244–2249. doi: 10.1109/PIMRC.2010.5671686.

[7] A. A. Khan, M. H. Rehmani, and A. Rachedi, “Cognitive-Radio-Based Internet of Things: Applications, Architectures, Spectrum Related Functionalities, and Future Research Directions,” *IEEE Wirel. Commun.*, vol. 24, no. 3, pp. 17–25, Jun. 2017, doi: 10.1109/MWC.2017.1600404.

[8] W. Wang, A. Kwasinski, D. Niyato, and Z. Han, “A Survey on Applications of Model-Free Strategy Learning in Cognitive Wireless Networks,” *IEEE Commun. Surv. Tutorials*, vol. 18, no. 3, pp. 1717–1757, 2016, doi: 10.1109/COMST.2016.2539923.

[9] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction (2nd Edition)*. The MIT Press, Cambridge, Massachusetts, 2018.

[10] R. S. Sutton *et al.*, “Fast gradient-descent methods for temporal-difference learning with linear function approximation,” *ACM Int. Conf. Proceeding Ser.*, vol. 382, pp. 993–1000, 2009, doi: 10.1145/1553374.1553501.

[11] K. Zhang, Z. Yang, and T. Başar, “Multi-Agent Reinforcement Learning: A Selective Overview of Theories and Algorithms,” Nov. 2019, [Online]. Available: http://arxiv.org/abs/1911.10635

[12] M. S. Stanković and S. S. Stanković, “Multi-agent temporal-difference learning with linear function approximation: Weak convergence under time-varying network topologies,” *Proc. Am. Control Conf.*, vol. 2016-July, pp. 167–172, 2016, doi: 10.1109/ACC.2016.7524910.

[13] M. S. Stankovic, M. Beko, and S. S. Stankovic, “Distributed gradient temporal difference off-policy learning with eligibility traces: Weak convergence,” in *Accepted at IFAC World Congress*, 2020.

[14] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. The MIT Press, Cambridge, Massachusetts, 2016. [Online]. Available:

https://lccn.loc.gov/2016022992
[15] C. Zhang, P. Patras, and H. Haddadi, “Deep Learning in Mobile and Wireless Networking: A Survey,” *IEEE Commun. Surv. Tutorials*, vol. 21, no. 3, pp. 2224–2287, 2019, doi: 10.1109/COMST.2019.2904897.
[16] M. A. Alsheikh, D. Niyato, S. Lin, H. Tan, and Z. Han, “Mobile Big Data Analytics Using Deep Learning and Apache Spark,” *IEEE Netw.*, vol. 30, no. June, pp. 22–29, 2016.
[17] Pedro Domingos, “A Few Useful Things to Know about Machine Learning,” *Commun. ACM*, vol. 55, no. 10, pp. 79–88, 2012, [Online]. Available: https://dl.acm.org/citation.cfm?id=2347755
[18] A. Yousefpour *et al.*, “All one needs to know about fog computing and related edge computing paradigms: A complete survey,” *J. Syst. Archit.*, vol. 98, no. September 2018, pp. 289–330, 2019, doi: 10.1016/j.sysarc.2019.02.009.
[19] C. Wu *et al.*, “Spectrum Management of Cognitive Radio Using Multi-agent Reinforcement Learning General Terms Performance, Experimentation Cite as: Spectrum Management of Cognitive Radio Using Multi-agent Reinforcement Learning”, [Online]. Available: www.ifaamas.org
[20] T. Jiang, D. Grace, and P. D. Mitchell, “Efficient exploration in reinforcement learning-based cognitive radio spectrum sharing,” *IET Commun.*, vol. 5, no. 10, pp. 1309–1317, 2011, doi: 10.1049/iet-com.2010.0258.
[21] Z. Tian, J. Wang, J. Wang, and J. Song, “Distributed NOMA-Based Multi-Armed Bandit Approach for Channel Access in Cognitive Radio Networks,” *IEEE Wirel. Commun. Lett.*, vol. 8, no. 4, pp. 1112–1115, 2019, doi: 10.1109/LWC.2019.2908371.
[22] N. Modi, P. Mary, and C. Moy, “QoS Driven Channel Selection Algorithm for Cognitive Radio Network: Multi-User Multi-Armed Bandit Approach,” *IEEE Trans. Cogn. Commun. Netw.*, vol. 3, no. 1, pp. 49–66, 2017, doi: 10.1109/TCCN.2017.2675901.
[23] I. Mustapha, B. M. Ali, A. Sali, M. F. A. Rasid, and H. Mohamad, “An energy efficient Reinforcement Learning based Cooperative Channel Sensing for Cognitive Radio Sensor Networks,” *Pervasive Mob. Comput.*, vol. 35, pp. 165–184, 2017, doi: 10.1016/j.pmcj.2016.07.007.
[24] W. Ning, X. Huang, K. Yang, F. Wu, and S. Leng, “Reinforcement learning enabled cooperative spectrum sensing in cognitive radio networks,” *J. Commun. Networks*, vol. 22, no. 1, pp. 12–22, 2020, doi: 10.1109/JCN.2019.000052.
[25] A. Kaur and K. Kumar, “Imperfect CSI Based Intelligent Dynamic Spectrum Management Using Cooperative Reinforcement Learning Framework in Cognitive Radio Networks,” *IEEE Trans. Mob. Comput.*, vol. 21, no. 5, pp. 1672–1683, 2022, doi: 10.1109/TMC.2020.3026415.
[26] S. J. Jang, C. H. Han, K. E. Lee, and S. J. Yoo, “Reinforcement learning-based dynamic band and channel selection in cognitive radio ad-hoc networks,” *Eurasip J. Wirel. Commun. Netw.*, vol. 2019, no. 1, pp. 1–25, 2019, doi: 10.1186/s13638-019-1433-1.
[27] R. Naparstek and K. Cohen, “Deep Multi-User Reinforcement Learning for Distributed Dynamic Spectrum Access,” *IEEE Trans. Wirel. Commun.*, vol. 18, no. 1, pp. 310–323, 2019, doi: 10.1109/TWC.2018.2879433.
[28] S. Wang, H. Liu, P. H. Gomes, and B. Krishnamachari, “Deep Reinforcement Learning for Dynamic Multichannel Access in Wireless Networks,” *IEEE Trans. Cogn. Commun. Netw.*, vol. 4, no. 2, pp. 257–265, 2018, doi: 10.1109/tccn.2018.2809722.

[29] V. Raj, I. Dias, T. Tholeti, and S. Kalyani, “Spectrum Access in Cognitive Radio Using a Two-Stage Reinforcement Learning Approach,” *IEEE J. Sel. Top. Signal Process.*, vol. 12, no. 1, pp. 20–34, 2018, doi: 10.1109/JSTSP.2018.2798920.

[30] Y. Lin, C. Wang, J. Wang, and Z. Dou, “A novel dynamic spectrum access framework based on reinforcement learning for cognitive radio sensor networks,” *Sensors (Switzerland)*, vol. 16, no. 10, 2016, doi: 10.3390/s16101675.

[31] A. Kaur and K. Kumar, “Intelligent spectrum management based on reinforcement learning schemes in cooperative cognitive radio networks,” *Phys. Commun.*, vol. 43, 2020, doi: 10.1016/j.phycom.2020.101226.

[32] P. Yang *et al.*, “Dynamic Spectrum Access in Cognitive Radio Networks Using Deep Reinforcement Learning and Evolutionary Game,” *2018 IEEE/CIC Int. Conf. Commun. China, ICCC 2018*, no. Iccc, pp. 405–409, 2018, doi: 10.1109/ICCChina.2018.8641242.

[33] Z. Li, F. R. Yu, and M. Huang, “A distributed consensus-based cooperative spectrum-sensing scheme in cognitive radios,” *IEEE Trans. Veh. Technol.*, vol. 59, no. 1, pp. 383–393, 2010, doi: 10.1109/TVT.2009.2031181.

[34] X. Tan *et al.*, “Cooperative Multi-Agent Reinforcement-Learning-Based Distributed Dynamic Spectrum Access in Cognitive Radio Networks,” *IEEE Internet Things J.*, vol. 9, no. 19, pp. 19477–19488, 2022, doi: 10.1109/JIOT.2022.3168296.

[35] A. Gao, C. Du, S. X. Ng, and W. Liang, “A Cooperative Spectrum Sensing with Multi-Agent Reinforcement Learning Approach in Cognitive Radio Networks,” *IEEE Commun. Lett.*, vol. 25, no. 8, pp. 2604–2608, 2021, doi: 10.1109/LCOMM.2021.3078442.

[36] A. Gao *et al.*, “Attention Enhanced Multi-Agent Reinforcement Learning for Cooperative Spectrum Sensing in Cognitive Radio Networks,” *IEEE Trans. Veh. Technol.*, vol. 73, no. 7, pp. 10464–10477, 2024, doi: 10.1109/TVT.2024.3384393.

[37] T. J. O’Shea and N. West, “Radio Machine Learning Dataset Generation with GNU Radio,” *6th GNU Radio Conf.*, pp. 1–6, 2016.

[38] T. J. O’Shea, J. Corgan, and T. C. Clancy, “Convolutional radio modulation recognition networks,” in *International Conference on Engineering Applications of Neural Networks, EANN 2016*, 2016, pp. 213–226. doi: 10.1007/978-3-319-44188-7_16.

[39] Q. Wen *et al.*, “Time Series Data Augmentation for Deep Learning: A Survey,” *arXiv*, vol. abs/2002.1, 2020, [Online]. Available: http://arxiv.org/abs/2002.12478

[40] N. E. West and T. O’Shea, “Deep architectures for modulation recognition,” in *2017 IEEE International Symposium on Dynamic Spectrum Access Networks, DySPAN 2017*, 2017. doi: 10.1109/DySPAN.2017.7920754.

[41] D. Perera, C., Zaslavsky, A., Christen, P., Georgakopoulos, “Context aware computing for the internet of things: A survey,” *IEEE Commun. Surv. Tutorials*, vol. 16, no. 1, pp. 414 – 454, 2014.

[42] D. Dašić, M. Vučetić, M. Perić, M. Beko, and M. Stanković, “Cooperative Multi-Agent Reinforcement Learning for Spectrum Management in IoT Cognitive Networks,” in *Proceedings of the 10th International Conference on Web Intelligence, Mining and Semantics*, Biarritz, France: Association for Computing Machinery, New York, NY, USA, 2020, pp. 238–247. doi: 10.1145/3405962.3405996.

[43] D. Dašić, N. Ilić, M. Vučetić, M. Perić, M. Beko, and M. S. Stanković, “Distributed

Spectrum Management in Cognitive Radio Networks by Consensus-Based Reinforcement Learning," *Sensors*, vol. 21, no. Machine Learning Applied to Sensor Data Analysis, pp. 1–20, 2021, doi: 10.3390/s21092970.

[44] M. S. Stankovic, M. Beko, and S. S. Stankovic, "Distributed Value Function Approximation for Collaborative Multi-Agent Reinforcement Learning," *IEEE Trans. Control Netw. Syst.*, pp. 1–11, 2021, doi: 10.1109/TCNS.2021.3061909.

[45] S. Valcarcel Macua, J. Chen, S. Zazo, and A. H. Sayed, "Distributed Policy Evaluation Under Multiple Behavior Strategies," *IEEE Trans. Automat. Contr.*, vol. 60, no. 5, pp. 1260–1274, May 2015, doi: 10.1109/TAC.2014.2368731.

[46] A. A. Khan, M. H. Rehmani, and A. Rachedi, "When Cognitive Radio meets the Internet of Things?," *2016 Int. Wirel. Commun. Mob. Comput. Conf. IWCMC 2016*, pp. 469–474, 2016, doi: 10.1109/IWCMC.2016.7577103.

[47] Davis *et al.*, "Ericsson Mobility Report Letter," 2023.

[48] J. Mitola and G. Q. Maguire, "Cognitive radio: making software radios more personal," *IEEE Pers. Commun.*, vol. 6, no. 4, pp. 13–18, 1999, doi: 10.1109/98.788210.

[49] P. Rawat, K. D. Singh, and J. M. Bonnin, "Cognitive radio for M2M and Internet of Things: A survey," *Comput. Commun.*, vol. 94, pp. 1–29, 2016, doi: 10.1016/j.comcom.2016.07.012.

[50] A. Raschellà, L. Militano, G. Araniti, A. Orsino, and A. Iera, "Cognitive Management Strategies for Dynamic Spectrum Access," W. Zhang, Ed., Singapore: Springer Singapore, 2019, pp. 533–567. doi: 10.1007/978-981-10-7594-2_16.

[51] K. C. Chen, Y. J. Peng, N. Prasad, Y. C. Liang, and S. Sun, "Cognitive radio network architecture: Part I - General structure," *Proc. 2nd Int. Conf. Ubiquitous Inf. Manag. Commun. ICUIMC-2008*, no. January, pp. 114–119, 2008, doi: 10.1145/1352793.1352817.

[52] V. Balaji, P. Kabra, P. V. P. K. Saieesh, C. Hota, and G. Raghurama, "Cooperative Spectrum Sensing in Cognitive Radios Using Perceptron Learning for IEEE 802.22 WRAN," *Procedia Comput. Sci.*, vol. 54, pp. 14–23, 2015, doi: 10.1016/j.procs.2015.06.002.

[53] A. Khattab and M. A. Bayoumi, "Standardization of cognitive radio networking: a comprehensive survey," *Ann. des Telecommun. Telecommun.*, vol. 70, no. 11–12, pp. 465–477, 2015, doi: 10.1007/s12243-015-0468-5.

[54] K. Ishizu, K. Mizutani, T. Matsumura, Z. Lan, and H. Harada, "IEEE 802.11af Wi-Fi in TV White Space," in *Handbook of Cognitive Radio*, W. Zhang, Ed., Singapore: Springer Singapore, 2019, pp. 1509–1535. doi: 10.1007/978-981-1389-8_53-1.

[55] O. Holland *et al.*, "The IEEE Dynamic Spectrum Access Networks Standards Committee (DySPAN-SC) and IEEE 1900 Working Groups," in *Opportunistic Spectrum Sharing and White Space Access: The Practical Reality*, O. Holland, H. Bogucka, and A. Medeisis, Eds., Wiley Telecom, 2015, pp. 631–648. doi: 10.1002/9781119057246.ch25.

[56] V. Mnih *et al.*, "Asynchronous methods for deep reinforcement learning," *33rd Int. Conf. Mach. Learn. ICML 2016*, vol. 4, pp. 2850–2869, 2016.

[57] Z. Liu, X. Xu, P. Qiao, and D. Li, "Acceleration for Deep Reinforcement Learning using Parallel and Distributed Computing: A Survey," *ACM Comput. Surv.*, vol. 19, no. 4, 2024, doi: 10.1145/3703453.

[58] R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, and I. Mordatch, "Multi-agent actor-critic for mixed cooperative-competitive environments," in *Proceedings of the*

31st International Conference on Neural Information Processing Systems, in NIPS'17. Red Hook, NY, USA: Curran Associates Inc., 2017, pp. 6382–6393.

[59] J. Foerster, R. Y. Chen, M. Al-Shedivat, S. Whiteson, P. Abbeel, and I. Mordatch, "Learning with Opponent-Learning Awareness," in *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, in AAMAS '18. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems, 2018, pp. 122–130.

[60] A. Nowé, P. Vrancx, and Y.-M. De Hauwere, "Game Theory and Multi-agent Reinforcement Learning," in *Reinforcement Learning: State-of-the-Art*, M. Wiering and M. van Otterlo, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 441–470. doi: 10.1007/978-3-642-27645-3_14.

[61] K. Zhang, Z. Yang, and T. Ba\csar, "Multi-Agent Reinforcement Learning: A Selective Overview of Theories and Algorithms," in *Handbook of Reinforcement Learning and Control*, K. G. Vamvoudakis, Y. Wan, F. L. Lewis, and D. Cansever, Eds., Cham: Springer International Publishing, 2021, pp. 321–384. doi: 10.1007/978-3-030-60990-0_12.

[62] L. Busoniu, R. Babuska, and B. De Schutter, "A Comprehensive Survey of Multiagent Reinforcement Learning," *IEEE Trans. Syst. Man, Cybern. Part C (Applications Rev.*, vol. 38, no. 2, pp. 156–172, 2008, doi: 10.1109/TSMCC.2007.913919.

[63] S. Gronauer and K. Diepold, "Multi-agent deep reinforcement learning: a survey," *Artif. Intell. Rev.*, vol. 55, no. 2, pp. 895–943, 2022, doi: 10.1007/s10462-021-09996-w.

[64] M. S. Stanković, M. Beko, and S. S. Stanković, "Distributed consensus-based multi-agent temporal-difference learning," *Automatica*, vol. 151, p. 110922, 2023, doi: 10.1016/j.automatica.2023.110922.

[65] H. Yu, "On Convergence of some Gradient-based Temporal-Differences Algorithms for Off-Policy Learning," pp. 1–86, 2017, [Online]. Available: http://arxiv.org/abs/1712.09652

[66] H. Yu, A. R. Mahmood, and R. S. Sutton, "On generalized Bellman equations and temporal-difference learning," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 10233 LNAI, pp. 3–14, 2017, doi: 10.1007/978-3-319-57351-9_1.

[67] C. Godsil and G. F. Royle, *Algebraic Graph Theory*, no. Book 207. in Graduate Texts in Mathematics. Springer, 2001. doi: 10.1007/978-1-4613-0163-9.

[68] M. S. Stanković, N. Ilić, and S. S. Stanković, "Distributed Stochastic Approximation: Weak Convergence and Network Design," *IEEE Trans. Automat. Contr.*, vol. 61, no. 12, pp. 4069–4074, 2016, doi: 10.1109/TAC.2016.2545098.

[69] H. J. Kushner and G. Yin, "Asymptotic Properties of Distributed and Communicating Stochastic Approximation Algorithms.," *SIAM J. Control Optim.*, vol. 25, no. 5, pp. 1266–1290, 1987, doi: 10.1137/0325070.

[70] H. Kushner and G. G. Yin, *Stochastic Approximation and Recursive Algorithms and Applications*. in Stochastic Modelling and Applied Probability. Springer New York, 2003. [Online]. Available: https://books.google.rs/books?id=_0blieuUJGkC

[71] C. Jiang, H. Zhang, Y. Ren, Z. Han, K.-C. Chen, and L. Hanzo, "Machine Learning Paradigms for Next-Generation Wireless Networks," *IEEE Wirel. Commun.*, vol. 24, pp. 98–105, 2017, doi: 10.1109/MWC.2016.1500356WC.

[72] Z. H. Zhou and J. Feng, "Deep forest: Towards an alternative to deep neural networks," *IJCAI Int. Jt. Conf. Artif. Intell.*, vol. 0, pp. 3553–3559, 2017, doi:

.24963/ijcai.2017/497.
[73] A. C. Damianou and N. D. Lawrence, “Deep Gaussian Processes,” *Proc. 16th Int. Conf. Artif. Intell. Stat.*, vol. 31, no. 3, pp. 514–521, 2013, [Online]. Available: http://proceedings.mlr.press/v31/damianou13a.pdf
[74] R. Collobert and S. Bengio, “Links between Perceptrons, MLPs and SVMs,” in *Proceedings of the 21st International Conference on Machine Learning, ICML 2004*, 2004, pp. 177–184. doi: 10.1145/1015330.1015415.
[75] B. A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,” *Commun. ACM*, vol. 60, no. 6, pp. 84–90, 2012, doi: 10.1145/3065386.
[76] G. E. Hinton, S. Osindero, and Y.-W. Teh, “A Fast Learning Algorithm for Deep Belief Nets,” *Neural Comput.*, vol. 18, pp. 1527–1554, 2006, doi: 10.1162/neco.2006.18.7.1527.
[77] Y. Bengio, “Learning deep architectures for AI,” *Found. Trends Mach. Learn.*, vol. 2, no. 1, pp. 1–127, 2009, doi: 10.1561/2200000006.
[78] C. Doersch, “Tutorial on Variational Autoencoders,” 2021. [Online]. Available: http://arxiv.org/abs/1606.05908
[79] I. Sutskever, O. Vinyals, and Q. V. Le, “Sequence to sequence learning with neural networks,” *Adv. Neural Inf. Process. Syst.*, vol. 4, no. January, pp. 3104–3112, 2014.
[80] F. A. Gers, J. Schmidhuber, and F. Cummins, “Learning to forget: Continual prediction with LSTM,” in *9th International Conference on Artificial Neural Networks, ICANN ’99*, 1999, pp. 850–855. doi: 10.1049/cp:19991218.
[81] K. Cho *et al.*, “Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation,” in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1724–1734. doi: 10.3115/v1/D14-1179.
[82] T. Wolf *et al.*, “Transformers : State-of-the-Art Natural Language Processing,” in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Systems Demonstrations*, Association for Computational Linguistics, 2020, pp. 38–45. doi: 10.18653/v1/2020.emnlp-demos.6.
[83] A. Vaswani *et al.*, “Attention is All you Need,” in *Advances in Neural Information Processing Systems (NIPS 2017)*, Curran Associates, Inc., 2017, pp. 5998–6008. doi: 10.1109/2943.974352.
[84] I. J. Goodfellow, J. Shlens, and C. Szegedy, “Explaining and harnessing adversarial examples,” in *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 2015, pp. 1–11.
[85] M. Arjovsky, S. Chintala, and L. Bottou, “Wasserstein generative adversarial networks,” in *34th International Conference on Machine Learning, PMLR 70*, 2017, pp. 214–223.
[86] M. Polese, R. Jana, V. Kounev, K. Zhang, S. Deb, and M. Zorzi, “Machine Learning at the Edge: A Data-Driven Architecture with Applications to 5G Cellular Networks,” *IEEE Trans. Mob. Comput.*, 2020, doi: 10.1109/tmc.2020.2999852.
[87] M. McClellan, C. Cervelló-Pastor, and S. Sallent, “Deep learning at the mobile edge: Opportunities for 5G networks,” *Appl. Sci.*, vol. 10, no. 14, 2020, doi: 10.3390/app10144735.
[88] L. Fernández Maimó, A. Huertas Celdrán, M. Gil Pérez, F. J. García Clemente, and G. Martínez Pérez, “Dynamic management of a deep learning-based anomaly detection system for 5G networks,” *J. Ambient Intell. Humaniz. Comput.*, vol. 10, no. 8, pp. 3083–3097, 2019, doi: 10.1007/s12652-018-0813-

[89] X. Zhao *et al.*, "Deep learning based mobile data offloading in mobile edge computing systems," *Futur. Gener. Comput. Syst.*, vol. 99, pp. 346–355, 2019, doi: 10.1016/j.future.2019.04.039.
[90] M. Mehrabi, M. Mohammadkarimi, M. Ardakani, and Y. Jing, "Decision Directed Channel Estimation Based on Deep Neural Network k-Step Predictor for MIMO Communications in 5G," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 11, pp. 2443–2456, 2019, doi: 10.1109/JSAC.2019.2934004.
[91] P. Hu and J. Zhang, "5G-Enabled Fault Detection and Diagnostics: How Do We Achieve Efficiency?," *IEEE Internet Things J.*, vol. 7, no. 4, pp. 3267–3281, 2020, doi: 10.1109/JIOT.2020.2965034.
[92] C. Zhong, M. C. Gursoy, and S. Velipasalar, "A deep reinforcement learning-based framework for content caching," *2018 52nd Annu. Conf. Inf. Sci. Syst. CISS 2018*, pp. 1–6, 2018, doi: 10.1109/CISS.2018.8362276.
[93] B. Serbetci and J. Goseling, "On optimal geographical caching in heterogeneous cellular networks," *IEEE Wireless Communications and Networking Conference, WCNC*, 2017, pp. 1–6. doi: 10.1109/WCNC.2017.7925847.
[94] B. Hussain, Q. Du, S. Zhang, A. Imran, and M. A. Imran, "Mobile edge computing-based data-driven deep learning framework for anomaly detection," *IEEE Access*, vol. 7, pp. 137656–137667, 2019, doi: 10.1109/ACCESS.2019.2942485.
[95] H. Abdah, J. P. Barraca, and R. L. Aguiar, "Handover Prediction Integrated with Service Migration in 5G Systems," in *IEEE International Conference on Communications*, 2020, pp. 1–7. doi: 10.1109/ICC40277.2020.9149426.
[96] J. Wang, L. Zhao, J. Liu, and N. Kato, "Smart Resource Allocation for Mobile Edge Computing: A Deep Reinforcement Learning Approach," *IEEE Trans. Emerg. Top. Comput.*, 2019, doi: 10.1109/tetc.2019.2902661.
[97] A. Ignatov *et al.*, "AI benchmark: All about deep learning on smartphones in 2019," in *Proceedings - 2019 International Conference on Computer Vision Workshop, ICCVW 2019*, IEEE Computer Society, 2019, pp. 3617–3635. doi: 10.1109/ICCVW.2019.00447.
[98] D. Z. Yazti and S. Krishnaswamy, "Mobile big data analytics: Research, practice, and opportunities," *Proc. - IEEE Int. Conf. Mob. Data Manag.*, vol. 1, pp. 1–2, 2014, doi: 10.1109/MDM.2014.73.
[99] D. Naboulsi, M. Fiore, S. Ribot, and R. Stanica, "Large-scale mobile traffic analysis: A survey," *IEEE Commun. Surv. Tutorials*, vol. 18, no. 1, pp. 124–161, 2016, doi: 10.1109/COMST.2015.2491361.
[100] C. Zhang, X. Ouyang, and P. Patras, "ZipNet-GAN: Inferring fine-grained mobile traffic patterns via a generative adversarial neural network," *Conex. 2017 - Proc. 2017 13th Int. Conf. Emerg. Netw. Exp. Technol.*, pp. 363–375, 2017, doi: 10.1145/3143361.3143393.
[101] L. Pierucci and D. Micheli, "A Neural Network for Quality of Experience Estimation in Mobile Communications," *IEEE Multimed.*, vol. 23, no. 4, pp. 42–49, 2016, doi: 10.1109/MMUL.2016.21.
[102] J. Wang *et al.*, "Spatiotemporal modeling and prediction in cellular networks: A big data enabled deep learning approach," in *Proceedings - IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, 2017, pp. 1–9. doi: 10.1109/INFOCOM.2017.8057090.
[103] C. W. Huang, C. T. Chiang, and Q. Li, "A study of deep learning networks on mobile traffic forecasting," in *IEEE International Symposium on Personal, Indoor*

and Mobile Radio Communications, PIMRC, 2018, pp. 1–6. doi: 10.1109/PIMRC.2017.8292737.

[104] J. Feng, X. Chen, R. Gao, M. Zeng, and Y. Li, "DeepTP: An End-to-End Neural Network for Mobile Cellular Traffic Prediction," *IEEE Netw.*, vol. 32, no. 6, pp. 108–115, 2018, doi: 10.1109/MNET.2018.1800127.

[105] I. Alawe, A. Ksentini, Y. Hadjadj-Aoul, and P. Bertin, "Improving Traffic Forecasting for 5G Core Network Scalability: A Machine Learning Approach," *IEEE Netw.*, vol. 32, no. 6, pp. 42–49, 2018, doi: 10.1109/MNET.2018.1800104.

[106] L. Chen, D. Yang, D. Zhang, C. Wang, J. Li, and T. M. T. Nguyen, "Deep mobile traffic forecast and complementary base station clustering for C-RAN optimization," *J. Netw. Comput. Appl.*, vol. 121, pp. 59–69, 2018, doi: 10.1016/j.jnca.2018.07.015.

[107] Z. Wang, "The Applications of Deep Learning on Traffic Identification," *Black Hat USA*, 2015.

[108] G. Aceto, D. Ciuonzo, A. Montieri, and A. Pescapé, "Mobile encrypted traffic classification using deep learning: Experimental evaluation, lessons learned, and challenges," *IEEE Trans. Netw. Serv. Manag.*, vol. 16, no. 2, pp. 445–458, 2019, doi: 10.1109/TNSM.2019.2899085.

[109] W. Wang, M. Zhu, J. Wang, X. Zeng, and Z. Yang, "End-To-end encrypted traffic classification with one-dimensional convolution neural networks," in *2017 IEEE International Conference on Intelligence and Security Informatics: Security and Big Data, ISI 2017*, 2017, pp. 43–48. doi: 10.1109/ISI.2017.8004872.

[110] W. Wang, M. Zhu, X. Zeng, X. Ye, and Y. Sheng, "Malware traffic classification using convolutional neural network for representation learning," in *International Conference on Information Networking*, 2017, pp. 712–717. doi: 10.1109/ICOIN.2017.7899588.

[111] V. C. Liang *et al.*, "Mercury: Metro density prediction with recurrent neural network on streaming CDR data," in *2016 IEEE 32nd International Conference on Data Engineering, ICDE 2016*, 2016, pp. 1374–1377. doi: 10.1109/ICDE.2016.7498348.

[112] B. Felbo, P. Sundsøy, A. Pentland, S. Lehmann, and Y.-A. de Montjoye, "Using Deep Learning to Predict Demographics from Mobile Phone Metadata," in *Proceedings of workshop track of the International Conference on Learning Representations (ICLR), 2016*, 2016, pp. 1–16.

[113] X. Cheng, L. Fang, X. Hong, and L. Yang, "Exploiting Mobile Big Data: Sources, Features, and Applications," *IEEE Netw.*, vol. 31, no. 1, pp. 72–79, 2017, doi: 10.1109/MNET.2017.1500295NM.

[114] K. Zheng, Z. Yang, K. Zhang, P. Chatzimisios, K. Yang, and W. Xiang, "Big data-driven optimization for mobile networks toward 5G," *IEEE Netw.*, vol. 30, no. 1, pp. 44–51, 2016, doi: 10.1109/MNET.2016.7389830.

[115] S. Liu and J. Du, "Poster: MobiEar-Building an Environment-Independent Acoustic Sensing Platform for the Deaf Using Deep Learning," in *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services Companion*, in MobiSys '16 Companion. New York, NY, USA: Association for Computing Machinery, 2016. doi: 10.1145/2938559.2948831.

[116] L. Tobias, A. Ducournau, F. Rousseau, G. Mercier, and R. Fablet, "Convolutional Neural Networks for Object Recognition on Mobile Devices: A Case Study," *Proc. - 23rd Int. Conf. Pattern Recognit.*, pp. 3530–3535, 2016, doi: 10.1109/ICPR.2016.7900181.

[117] B. Almaslukh, J. Almuhtadi, and A. Artoli, "An Effective Deep Autoencoder

Approach for Online Smartphone-Based Human Activity Recognition," *Int. J. Comput. Sci. Netw. Secur.*, vol. 17, no. 4, pp. 160–165, 2017, [Online]. Available: http://www.researchgate.net/publication/323019783

[118] S. Bhattacharya and N. D. Lane, "From smart to deep: Robust activity recognition on smartwatches using deep learning," in *2016 IEEE International Conference on Pervasive Computing and Communication Workshops, PerCom Workshops 2016*, IEEE, 2016. doi: 10.1109/PERCOMW.2016.7457169.

[119] M. Zhao *et al.*, "RF-based 3D skeletons," *SIGCOMM 2018 - Proc. 2018 Conf. ACM Spec. Interes. Gr. Data Commun.*, pp. 267–281, 2018, doi: 10.1145/3230543.3230579.

[120] T. Yoshioka *et al.*, "The NTT CHiME-3 System: Advances in Speech Enhancement and Recognition for Mobile Multi-microphone Devices," in *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2015, pp. 436–443. doi: 10.1109/ASRU.2015.7404828.

[121] P. Georgiev, S. Bhattacharya, N. D. Lane, and C. Mascolo, "Low-resource Multi-task Audio Sensing for Mobile and Embedded Devices via Shared Deep Neural Network Representations," in *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2017, pp. 1–19. doi: 10.1145/3131895.

[122] X. Song, K. Hiroshi, and S. Ryosuke, "DeepTransport: Prediction and Simulation of Human Mobility and Transportation Mode at a Citywide Level," in *IJCAI International Joint Conference on Artificial Intelligence*, 2016, pp. 2618–2624.

[123] R. Jiang *et al.*, "DeepUrbanMomentum: An online deep-learning system for short-term urban mobility prediction," in *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*, 2018, pp. 784–791.

[124] A. Shokry, M. Torki, and M. Youssef, "DeepLoc: A Ubiquitous Accurate and Low-Overhead Outdoor Cellular Localization System," in *GIS: Proceedings of the 26th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 2018, pp. 339–348. doi: 10.1145/3274895.3274909.

[125] Y. Zhou, Z. M. Fadlullah, B. Mao, and N. Kato, "A Deep-Learning-Based Radio Resource Assignment Technique for 5G Ultra Dense Networks," *IEEE Netw.*, vol. 32, no. 6, pp. 28–34, 2018, doi: 10.1109/MNET.2018.1800085.

[126] H. Rutagemwa, A. Ghasemi, and S. Liu, "Dynamic spectrum assignment for land mobile radio with deep recurrent neural networks," in *2018 IEEE International Conference on Communications Workshops, ICC Workshops 2018 - Proceedings*, IEEE, 2018, pp. 1–6. doi: 10.1109/ICCW.2018.8403659.

[127] X. Liu, Y. Xu, L. Jia, Q. Wu, and A. Anpalagan, "Anti-jamming communications using spectrum waterfall: A deep reinforcement learning approach," *IEEE Commun. Lett.*, vol. 22, no. 5, pp. 998–1001, 2018, doi: 10.1109/LCOMM.2018.2815018.

[128] Vrizlynn L. L. Thing, "IEEE 802.11 Network Anomaly Detection and Attack Classification: A Deep Learning Approach," in *IEEE Wireless Communications and Networking Conference (WCNC)*, San Francisco, CA, USA: IEEE, 2017. doi: 10.1109/WCNC.2017.7925567.

[129] M. E. Aminanto and K. Kim, "Detecting Impersonation Attack in WiFi Networks Using Deep Learning Approach," in *International Workshop on Information Security Applications, WISA 2016*, D. Choi and S. Guilley, Eds., Jeju Island, Korea: Springer International Publishing, 2017, pp. 136–147. doi: 10.1007/978-3-319-56549-1.

[130] Q. Feng, Y. Zhang, C. Li, Z. Dou, and J. Wang, "Anomaly Detection of Spectrum in Wireless Communication via Deep Auto-Encoders," *J. Supercomput.*, vol. 73,

. 7, pp. 3161–3178, 2017, doi: 10.1007/s11227-017-2017-7.
[131] Z. Yuan, Y. Lu, Z. Wang, and Y. Xue, “Droid-Sec: Deep learning in android malware detection,” *ACM SIGCOMM Comput. Commun. Rev.*, vol. 44, no. 4, 371–372, 2014, doi: 10.1145/2619239.2631434.
[132] N. McLaughlin *et al.*, “Deep android malware detection,” *CODASPY 2017 - Proc. 7th ACM Conf. Data Appl. Secur. Priv.*, pp. 301–308, 2017, doi: 10.1145/3029806.3023.
[133] H. Maghrebi, T. Portigliatti, and E. Prouff, “Breaking cryptographic implementations using deep learning techniques,” in *Security, Privacy, and Applied Cryptography Engineering, SPACE 2016*, C. Carlet, M. Anwar Hasan, and V. Saraswat, Eds., Springer, Cham, 2016, pp. 3–26. doi: 10.1007/978-3-9-49445-6_1.
[134] R. Ning, C. Wang, C. S. Xin, J. Li, and H. Wu, “DeepMag: Sniffing Mobile Apps in Magnetic Field through Deep Convolutional Neural Networks,” in *2018 IEEE International Conference on Pervasive Computing and Communications erCom)*, IEEE, 2018, pp. 1–10. doi: 10.1109/PERCOM.2018.8444573.
[135] N. Samuel, T. Diskin, and A. Wiesel, “Deep MIMO detection,” in *IEEE Workshop on Signal Processing Advances in Wireless Communications, SPAWC*, pporo, Japan: IEEE, 2017, pp. 1–5. doi: 10.1109/SPAWC.2017.8227772.
[136] M. A. Wijaya, K. Fukawa, and H. Suzuki, “Intercell-Interference Cancellation and Neural Network Transmit Power Optimization for MIMO Channels,” in *2015 IEEE 82nd Vehicular Technology Conference, VTC Fall 2015 - Proceedings*, Boston, A, USA: IEEE, 2015. doi: 10.1109/VTCFall.2015.7390988.
[137] T. J. O’Shea, T. Erpek, and T. C. Clancy, “Deep Learning Based MIMO Communications,” *ArXiv*, pp. 1–9, 2017, [Online]. Available: http://arxiv.org/abs/1707.07980
[138] A. Furno, M. Fiore, and R. Stanica, “Joint spatial and temporal classification of mobile traffic demands,” *Proc. - IEEE Conf. Comput. Commun. INFOCOM*, 17, doi: 10.1109/INFOCOM.2017.8057089.
[139] M. Geist and B. Scherrer, “Off-policy Learning With Eligibility Traces: A Survey,” *Mach. Learn. Res.*, vol. 15, pp. 289–333, 2014.
[140] K. Zhang, Z. Yang, and T. Başar, “Decentralized Multi-Agent Reinforcement Learning with Networked Agents: Recent Advances,” Dec. 2019, [Online]. vailable: http://arxiv.org/abs/1912.03821
[141] W. Y. Lee and I. F. Akyildiz, “Optimal spectrum sensing framework for cognitive radio networks,” *IEEE Trans. Wirel. Commun.*, vol. 7, no. 10, pp. 3845–3857, 2008, doi: 10.1109/T-WC.2008.070391.
[142] S. Kar, J. M. F. Moura, and H. V. Poor, *QD-learning: A collaborative distributed strategy for multi-agent reinforcement learning through consensus + novations*, vol. 61, no. 7. 2013. doi: 10.1109/TSP.2013.2241057.
[143] M. Di Felice, K. R. Chowdhury, C. Wu, L. Bononi, and W. Meleis, “Learning-based spectrum selection in Cognitive Radio Ad oc Networks,” in *Wired/Wireless Internet Communications. WWIC 2010.*, E. Osipov, A. Kassler, T. M. Bohnert, and X. Masip-Bruin, Eds., Springer, erlin, Heidelberg, 2010, pp. 33–145. doi: 10.1007/978-3-642-13315-2_11.
[144] K. O. Al Ali, N. Ilic, M. S. Stankovic, and S. S. Stankovic, “Distributed target tracking in sensor networks using multi-step consensus,” *IET Radar, Sonar vig.*, vol. 12, no. 9, pp. 998–1004, 2018, doi: 10.1049/iet-rsn.2018.5127.
[145] R. S. Sutton, A. R. Mahmood, and M. White, “An emphatic approach to the problem of off-policy temporal-difference learning,” *J. Mach. Learn. Res.*, vol.

pp. 1–29, 2016.
[146] J. Bhandari, D. Russo, R. Singal, S. Bubeck, V. Perchet, and P. Rigollet, "A Finite Time Analysis of Temporal Difference Learning With Linear Functi Approximation," in *Proceedings of 31st Conference On Learning Theory*, S. Bubeck, V. Perchet, and P. Rigollet, Eds., Stockholm, Sweden, 2018, pp. 1691–1692.
[147] A. Olshevsky and J. N. Tsitsiklis, "Convergence speed in distributed consensus and averaging," in *Proceedings of the 45th IEEE Conference on Decision and Control*, San Diego, CA, USA: IEEE, 2006, pp. 3387–3392. doi: 1109/CDC.2006.376899.
[148] Y. Saleem and M. H. Rehmani, "Primary radio user activity models for cognitive radio networks: A survey," *J. Netw. Comput. Appl.*, vol. 43, pp. 1–16, 2014, doi: .1016/j.jnca.2014.04.001.
[149] Y. Arjoune and N. Kaabouch, "A comprehensive survey on spectrum sensing in cognitive radio networks: Recent advances, new challenges, and future research directions," *Sensors (Switzerland)*, vol. 19, no. 1, 2019, doi: .3390/s19010126.
[150] H.-W. Kao and E. H.-K. Wu, "QoE Sustainability on 5G and Beyond 5G Networks," *IEEE Wirel. Commun.*, vol. 30, no. 1, pp. 118–125, 2023, doi: 1109/MWC.007.2200260.
[151] W. Chen, I. Demirkol, M. Mostafa, and S. Ruffini, "Series Editorial: Mobile Communications and Networks," *IEEE Commun. Mag.*, vol. 61, no. 4, pp. 26–2 2023, doi: 10.1109/MCOM.2023.10106153.
[152] "REPORT ITU-R M.2078 Estimated spectrum bandwidth requirements for the re development of IMT-2000 and IMT-Advanced," 2006.
[153] C. H. Lim, "Spectrum Sensing for a Radar Signal With Pulse Compression in Cognitive Radios," *IEEE Commun. Lett.*, vol. 26, no. 12, pp. 3074–3078, 2022, doi: 10.1109/LCOMM 22.3205613.
[154] M. Miantezila, J. Id, and B. Guo, "Sensing spectrum sharing sed massive MIMO radar for drone tracking and interception," *PLoS One*, pp. 1–18, 2022, : 10.1371/journal.pone.0268834.
[155] H. Rafiqi, G. Mahendru, and S. H. Gupta, "Effect of Relay-based Communication on Probability of Detection for Spectrum Sensing in LoRaWAN," *Wirel. Pers. Commun.*, vol. 130, no. 4, pp. 2345–2366, 2023, doi: 10.1007/s11277-023-273-y.
[156] L. Dai, R. Jiao, F. Adachi, H. V. Poor, and L. Hanzo, "Deep Learning for Wireless Communications: An Emerging Interdisciplinary Paradigm," *IEEE Wirel. mmun.*, vol. 27, no. 4, pp. 133–139, 2020, doi: 10.1109/MWC.001.1900491.
[157] D. Seo and H. Nam, "Deep RP-CNN for Burst Signal Detection in Cognitive dios," vol. 8, pp. 4–11, 2020, doi: 10.1109/ACCESS.2020.3023262.
[158] I. Alkhouri, M. Joneidi, F. Hejazi, and N. Rahnavard, "Large-Scale Spectrum Occupancy Learning via Tensor Decomposition and LSTM Networks," in *2020 IEEE International Radar Conference (RADAR)*, 2020, pp. 677–682. doi: .1109/RADAR42522.2020.9114785.
[159] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, "A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 33, no. 12, pp. 6999–7019, 2022, doi: 10. 9/TNNLS.2021.3084827.
[160] A. Sherstinsky, "Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network," *Phys. D Nonlinear Phenom.*, vol. 404, p.

2306, 2020, doi: https://doi.org/10.1016/j.physd.2019.132306.
[161] N. Balwani, D. K. Patel, B. Soni, and M. López-Benítez, "Long Short-Term Memory based Spectrum Sensing Scheme for Cognitive Radio," in *2019 IEEE 30th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, 2019, pp. 1–6. doi: 10.1109/PIMRC.2019.8904422.
[162] L. Yu, J. Chen, G. Ding, Y. Tu, J. Yang, and J. Sun, "Spectrum Prediction Based on Taguchi Method in Deep Learning With Long Short-Term Memory," *IEEE Access*, vol. 6, pp. 45923–45933, 2018, doi: 10.1109/ACCESS.2018.2864222.
[163] D. Dašić, M. Vučetić, N. Ilić, M. Stanković, and M. Beko, "Application of Deep Learning Algorithms and Architectures in the New Generation of Mobile Networks," *Serbian J. Electr. Eng.*, vol. 18, no. 3, pp. 397–426, 2021, doi: 10.2298/SJEE2103397D.

DISTRIBUTED REINFORCEMENT AND DEEP LEARNING FOR SPECTRUM MANAGEMENT IN COGNITIVE RADIO NETWORKS

ORIGINALITY REPORT

37%
SIMILARITY INDEX

PRIMARY SOURCES

1	www.mdpi.com Internet	2001 words — 5%
2	scindeks-clanci.ceon.rs Internet	938 words — 3%
3	Dejan Dašić, Miljan Vučetić, Miroslav Perić, Marko Beko, Miloš Stanković. "Cooperative Multi-Agent Reinforcement Learning for Spectrum Management in IoT Cognitive Networks", Proceedings of the 10th International Conference on Web Intelligence, Mining and Semantics, 2020 Crossref	892 words — 2%
4	Miloš S. Stanković, Marko Beko, Srdjan S. Stanković. "Distributed consensus-based multi-agent temporal-difference learning", Automatica, 2023 Crossref	771 words — 2%
5	portal.sinteza.singidunum.ac.rs Internet	543 words — 1%
6	www.ncbi.nlm.nih.gov Internet	351 words — 1%
7	arxiv.org Internet	326 words — 1%

8	Milos S. Stankovic, Marko Beko, Srdjan S. Stankovic. "Distributed Consensus-Based Multi-Agent Off-Policy Temporal-Difference Learning", 2021 60th IEEE Conference on Decision and Control (CDC), 2021 Crossref	228 words — 1%
9	mdpi-res.com Internet	223 words — 1%
10	ucilnica.fri.uni-lj.si Internet	223 words — 1%
11	export.arxiv.org Internet	117 words — < 1%
12	dokumen.pub Internet	115 words — < 1%
13	iris.poliba.it Internet	115 words — < 1%
14	www.etran.rs Internet	105 words — < 1%
15	Dejan Dašić, Nemanja Ilić, Miljan Vučetić, Miroslav Perić, Marko Beko, Miloš S. Stanković. "Distributed Spectrum Management in Cognitive Radio Networks by Consensus-Based Reinforcement Learning", Sensors, 2021 Crossref	101 words — < 1%
16	d-nb.info Internet	94 words — < 1%
17	www.researchgate.net Internet	92 words — < 1%

18	ebin.pub Internet	91 words — < 1%
19	nrl.northumbria.ac.uk Internet	86 words — < 1%
20	deepai.org Internet	83 words — < 1%
21	www.research.ed.ac.uk Internet	82 words — < 1%
22	Biswadip Basu Mallik, Gunjan Mukherjee, Rahul Kar, Aryan Chaudhary. "Deep Learning Concepts in Operations Research", Routledge, 2024 Publications	77 words — < 1%
23	Yingchun Wang, Jingyi Wang, Weizhan Zhang, Yufeng Zhan, Song Guo, Qinghua Zheng, Xuanyu Wang. "A survey on deploying mobile deep learning applications: A systemic and technical perspective", Digital Communications and Networks, 2022 Crossref	77 words — < 1%
24	theses.gla.ac.uk Internet	74 words — < 1%
25	Milos S. Stankovic, Marko Beko, Srdjan S. Stankovic. "Distributed Value Function Approximation for Collaborative Multiagent Reinforcement Learning", IEEE Transactions on Control of Network Systems, 2021 Crossref	71 words — < 1%
26	Dejan Dašić, Miljan Vučetić, Gardelito Hew A Kee, Miloš Stanković. "Deep Learning Applications in	66 words — < 1%

Mobile Networks", Proceedings of the International Scientific Conference - Sinteza 2019, 2019
Crossref

27 Vivek S. Sharma, Shubham Mahajan, Anand Nayyar, Amit Kant Pandit. "Deep Learning in Engineering, Energy and Finance - Principles and Applications", CRC Press, 2024 — 65 words — < 1%
Publications

28 mro.massey.ac.nz — 61 words — < 1%
Internet

29 aiforsocialgood.ca — 55 words — < 1%
Internet

30 etheses.whiterose.ac.uk — 54 words — < 1%
Internet

31 www.journal.uestc.edu.cn — 52 words — < 1%
Internet

32 repository.up.ac.za — 51 words — < 1%
Internet

33 Yixuan Zhang, Zhongqiang Luo. "A Review of Research on Spectrum Sensing Based on Deep Learning", Electronics, 2023 — 50 words — < 1%
Crossref

34 openaccess.altinbas.edu.tr — 50 words — < 1%
Internet

35 Uzair Aslam Bhatti, Jingbing Li, Mengxing Huang, Sibghat Ullah Bazai, Muhammad Aamir. "Deep Learning for Multimedia Processing Applications - Volume Two: Signal Processing and Pattern Recognition", CRC Press, 2024 — 48 words — < 1%
Publications

36	openprairie.sdstate.edu Internet	47 words — < 1%
37	nardus.mpn.gov.rs Internet	46 words — < 1%
38	Mudassar H. Naikwadi, Kishor P. Patil. "Spectrum Inference in Cognitive Radio Networks with Machine Learning", 2023 International Conference on Emerging Smart Computing and Informatics (ESCI), 2023 Crossref	44 words — < 1%
39	profpatel.in Internet	44 words — < 1%
40	Jingjing Wang, Chunxiao Jiang, Haijun Zhang, Yong Ren, Kwang-Cheng Chen, Lajos Hanzo. "Thirty Years of Machine Learning: The Road to Pareto-Optimal Wireless Networks", IEEE Communications Surveys & Tutorials, 2020 Crossref	42 words — < 1%
41	www.jatit.org Internet	42 words — < 1%
42	Partemie M. Mutescu, Alexandru Lavric, Adrian I. Petrariu, Valentin Popa. "Deep Learning Enhanced Spectrum Sensing for LoRa Spreading Factor Detection", 2023 13th International Symposium on Advanced Topics in Electrical Engineering (ATEE), 2023 Crossref	41 words — < 1%
43	Susandhika M, Shirly Edward A. "A Comprehensive Review and Comparative analysis of 5G and 6G based MIMO Channel Estimation Techniques",	41 words — < 1%

2023 International Conference on Recent Trends in Electronics and Communication (ICRTEC), 2023
Crossref

44	Vijeta Sharma, Manjari Gupta, Ajai Kumar, Deepti Mishra. "Video Processing using Deep learning Techniques: A Systematic Literature Review", IEEE Access, 2021 Crossref	41 words — < 1%
45	amslaurea.unibo.it Internet	41 words — < 1%
46	research-information.bris.ac.uk Internet	41 words — < 1%
47	Asmaa M. Aboelezz, Abdelfattah A. Eladl, Mohamed T. Abo Amasha, Manar H. Elderigi et al. "Efficient fault detection, localization, and isolation in MT-HVDC systems based on distance protection and LoRaWAN communication", Energy Reports, 2023 Crossref	40 words — < 1%
48	Ding Zhao, Zhan Zhang, Bin Yu, Yuehai Wang. "Time Domain Speech Enhancement using Self-Attention-Based Subspace Projection", 2021 7th International Conference on Computer and Communications (ICCC), 2021 Crossref	40 words — < 1%
49	vdoc.pub Internet	40 words — < 1%
50	www.univ-usto.dz Internet	40 words — < 1%
51	Dilip Kumar Baruah, Kuntala Boruah. "Early Detection Of Canine Babesia From Red Blood Cell	39 words — < 1%

Images Using Deep Ensemble Learning", Journal of Electronics, Electromedical Engineering, and Medical Informatics, 2024
Crossref

52 Nemanja Ilić, Dejan Dašić, Miljan Vučetić, Aleksej Makarov, Ranko Petrović. "Distributed web hacking by adaptive consensus-based reinforcement learning", Artificial Intelligence, 2024
Crossref
39 words — < 1%

53 Tao Zhang, Kun Zhu, Dusit Niyato. "Detection of Sleeping Cells in Self-organizing Cellular Networks: An Adversarial Auto-Encoder Method", IEEE Transactions on Cognitive Communications and Networking, 2021
Crossref
38 words — < 1%

54 bura.brunel.ac.uk
Internet
38 words — < 1%

55 "Handbook of Cognitive Radio", Springer Science and Business Media LLC, 2019
Crossref
37 words — < 1%

56 Amandeep Kaur, Krishan Kumar, Arun Prakash, Rajeev Tripathi. "Imperfect CSI-Based Resource Management in Cognitive IoT Networks: A Deep Recurrent Reinforcement Learning Framework", IEEE Transactions on Cognitive Communications and Networking, 2023
Crossref
37 words — < 1%

57 Sherief Hashima, Mostafa M. Fouda, Sadman Sakib, Zubair Md Fadlullah, Kohei Hatano, Ehab Mahmoud Mohamed, Xuemin Shen. "Energy-Aware Hybrid RF-VLC Multi-Band Selection in D2D Communication: A Stochastic Multi-Armed Bandit Approach", IEEE Internet of Things Journal, 2022
Crossref
37 words — < 1%

58	unidel.edu.ng Internet	37 words — < 1%
59	unsworks.unsw.edu.au Internet	37 words — < 1%
60	Manish Kumar Giri, Saikat Majumder. "Cooperative Spectrum Sensing Using Extreme Learning Machines for Cognitive Radio Networks", IETE Technical Review, 2021 Crossref	36 words — < 1%
61	discovery.ucl.ac.uk Internet	36 words — < 1%
62	eprints.nottingham.ac.uk Internet	36 words — < 1%
63	www.nowpublishers.com Internet	36 words — < 1%
64	"ICOA 2023 Conference Proceedings", 2023 9th International Conference on Optimization and Applications (ICOA), 2023 Crossref	35 words — < 1%
65	Ahmed Khattab, Dmitri Perkins, Magdy Bayoumi. "Cognitive Radio Networks", Springer Nature, 2013 Crossref	35 words — < 1%
66	Himanshu Agrawal, Krishna Asawa. "Decentralized Learning for Opportunistic Spectrum Access: Multiuser Restless Multiarmed Bandit Formulation", IEEE Systems Journal, 2019 Crossref	35 words — < 1%

67	Mohammad Asif Hossain, Rafidah Md Noor, Kok-Lim Alvin Yau, Saaidal Razalli Azzuhri, Muhammad Reza Zraba, Ismail Ahmedy. "Comprehensive Survey of Machine Learning Approaches in Cognitive Radio-based Vehicular Ad Hoc Networks", IEEE Access, 2020 Crossref	35 words — < 1%
68	Soodeh Amiri-Doomari, Ghasem Mirjalily, Jamshid Abouei. "Interference Aware Routing Game for Cognitive Radio Ad-hoc Networks", Journal of Telecommunications and Information Technology, 2018 Crossref	35 words — < 1%
69	Xiaowen Ye, Qian Zhou, Liqun Fu. "Deep Reinforcement Learning-Based Scheduling for NR-U/WiGig Coexistence in Unlicensed mmWave Bands", IEEE Transactions on Wireless Communications, 2024 Crossref	35 words — < 1%
70	Xuanheng Li, Yulong Zhang, Haichuan Ding, Yuguang Fang. "Intelligent Spectrum Sensing and Access with Partial Observation Based on Hierarchical Multi-Agent Deep Reinforcement Learning", IEEE Transactions on Wireless Communications, 2024 Crossref	35 words — < 1%
71	d197for5662m48.cloudfront.net Internet	35 words — < 1%
72	digital.lib.washington.edu Internet	35 words — < 1%
73	Akeel Bdrany, Sattar B. Sadkhan. "Decision Making Approaches in Cognitive Radio-Status, Challenges and Future Trends", 2020 International Conference on Advanced Science and Engineering (ICOASE), 2020 Crossref	34 words — < 1%

74 Amandeep Kaur, Krishan Kumar. "Imperfect CSI Based Intelligent Dynamic Spectrum Management Using Cooperative Reinforcement Learning Framework in Cognitive Radio Networks", IEEE Transactions on Mobile Computing, 2022
Crossref
34 words — < 1%

75 Ayesha Naeem, Mubashir Husain Rehmani, Yasir Saleem, Imran Rashid, Noel Crespi. "Network Coding in Cognitive Radio Networks: A Comprehensive Survey", IEEE Communications Surveys & Tutorials, 2017
Crossref
34 words — < 1%

76 Mostafa Al-Gabalawy, Nesreen S. Hosny, Ahmed R. Adly. "Probabilistic forecasting for energy time series considering uncertainties based on deep learning algorithms", Electric Power Systems Research, 2021
Crossref
34 words — < 1%

77 Raj Kumar Samanta, Bikash Sadhukhan, Hiranmay Samaddar, Suvobrata Sarkar, Chandan Koner, Monidepa Ghosh. "Scope of machine learning applications for addressing the challenges in next-generation wireless networks", CAAI Transactions on Intelligence Technology, 2022
Crossref
34 words — < 1%

78 Swarnava Dey, Arijit Mukherjee. "Implementing Deep Learning and Inferencing on Fog and Edge Computing Systems", 2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), 2018
Crossref
34 words — < 1%

79 repository.uantwerpen.be
Internet
34 words — < 1%

80	www.trudymai.ru Internet	34 words — < 1%
81	Abdullah Balcı, Radosveta Sokullu. "Grant Free NOMA with Deep Learning for Massive IoT Applications", 2023 International Balkan Conference on Communications and Networking (BalkanCom), 2023 Crossref	33 words — < 1%
82	Hamed Mosavat-Jahromi, Yue Li, Lin Cai, Jianping Pan. "Prediction and Modeling of Spectrum Occupancy for Dynamic Spectrum Access Systems", IEEE Transactions on Cognitive Communications and Networking, 2021 Crossref	33 words — < 1%
83	Samuel Dodge, Lina Karam. "Understanding how image quality affects deep neural networks", 2016 Eighth International Conference on Quality of Multimedia Experience (QoMEX), 2016 Crossref	33 words — < 1%
84	hal.archives-ouvertes.fr Internet	33 words — < 1%
85	id.123dok.com Internet	33 words — < 1%
86	publik.tuwien.ac.at Internet	33 words — < 1%
87	Abdulrahman Altahhan. "Self-reflective deep reinforcement learning", 2016 International Joint Conference on Neural Networks (IJCNN), 2016 Crossref	32 words — < 1%

88 An, Chunyan, Hong Ji, and Pengbo Si. "Dynamic Spectrum Access with QoS Provisioning in Cognitive Radio Networks", 2010 IEEE Global Telecommunications Conference GLOBECOM 2010, 2010.
Crossref
32 words — < 1%

89 Su P. Sone, Janne J. Lehtomaki, Zaheer Khan. "Wireless Traffic Usage Forecasting Using Real Enterprise Network Data: Analysis and Methods", IEEE Open Journal of the Communications Society, 2020
Crossref
32 words — < 1%

90 btu.edu.ge
Internet
32 words — < 1%

91 Arthur G.O. Mutambara. "Design and Analysis of Control Systems - Driving the Fourth Industrial Revolution", CRC Press, 2024
Publications
31 words — < 1%

92 Hugues M. Kamdjou, David Baudry, Vincent Havard, Samir Ouchani. "Resource-Constrained EXtended Reality Operated With Digital Twin in Industrial Internet of Things", IEEE Open Journal of the Communications Society, 2024
Crossref
31 words — < 1%

93 Wen Wu, Nan Cheng, Ning Zhang, Peng Yang, Khalid Aldubaikhy, Xuemin Shen. "Performance Analysis and Enhancement of Beamforming Training in 802.11ad", IEEE Transactions on Vehicular Technology, 2020
Crossref
31 words — < 1%

94 Xian Wu, Kun Xu, Peter Hall. "A survey of image synthesis and editing with generative adversarial networks", Tsinghua Science and Technology, 2017
Crossref
31 words — < 1%

95	csdl.computer.org Internet	31 words — < 1%
96	dspace.bits-pilani.ac.in:8080 Internet	31 words — < 1%
97	www.bastibl.net Internet	31 words — < 1%
98	Mario Santoro, Christine Nardini. "Large-scale analysis of the medical discourse on rheumatoid arthritis: complementing a socio-anthropologic analysis", Cold Spring Harbor Laboratory, 2024 Crossref Posted Content	30 words — < 1%
99	Milos S. Stankovic, Srdjan S. Stankovic. "Multi-agent temporal-difference learning with linear function approximation: Weak convergence under time-varying network topologies", 2016 American Control Conference (ACC), 2016 Crossref	30 words — < 1%
100	fastercapital.com Internet	30 words — < 1%
101	ijritcc.org Internet	30 words — < 1%
102	K. Krishna Kishore, Arun Sekar Rajasekaran, Ismail Keshta, Haewon Byeon, Mohammed Wasim Bhatt, Azeem Irshad, Isaac Ofori. "RETRACTED ARTICLE: Intelligent dynamic spectrum access using fuzzy logic in cognitive radio networks", Discover Applied Sciences, 2024 Crossref	29 words — < 1%
103	Mu Qiao, Haitao Zhao, Li Zhou, Chunsheng Zhu, Shengchun Huang. "Topology-Transparent	29 words — < 1%

Scheduling Based on Reinforcement Learning in Self-Organized Wireless Networks", IEEE Access, 2018
Crossref

104 Youssef Baddi, Mohammed Amin Almaiah, Omar Almomani, Yassine Maleh. "The Art of Cyber Defense - From Risk Assessment to Threat Intelligence", CRC Press, 2024 — 29 words — < 1%
Publications

105 scholarworks.utep.edu — 29 words — < 1%
Internet

106 Inam Ullah Khan, Mariya Ouaissa, Mariyam Ouaissa, Muhammad Fayaz, Rehmat Ullah. "Artificial Intelligence for Intelligent Systems - Fundamentals, Challenges, and Applications", CRC Press, 2024 — 28 words — < 1%
Publications

107 Jinzhong Yang, Gregory C. Sharp, Mark J. Gooding. "Auto-Segmentation for Radiation Oncology - State of the Art", CRC Press, 2021 — 28 words — < 1%
Publications

108 assets.researchsquare.com — 28 words — < 1%
Internet

109 dspace.lib.cranfield.ac.uk — 28 words — < 1%
Internet

110 ijimai.org — 28 words — < 1%
Internet

111 Niranjana Radhakrishnan, Sithamparanathan Kandeepan, Xinghuo Yu, Gianmarco Baldini. "Performance Analysis of Long Short-Term Memory-based Markovian Spectrum Prediction", IEEE Access, 2021 — 27 words — < 1%
Crossref

112	Syed Hashim Raza Bukhari, Mubashir Husain Rehmani, Sajid Siraj. "A Survey of Channel Bonding for Wireless Networks and Guidelines of Channel Bonding for Futuristic Cognitive Radio Sensor Networks", IEEE Communications Surveys & Tutorials, 2016 Crossref	27 words — < 1%
113	Syed Junaid Nawaz, Shree Krishna Sharma, Shurjeel Wyne, Mohammad N. Patwary, Md. Asaduzzaman. "Quantum Machine Learning for 6G Communication Networks: State-of-the-Art and Vision for the Future", IEEE Access, 2019 Crossref	27 words — < 1%
114	www.zora.uzh.ch Internet	27 words — < 1%
115	Gavrilovska, Liljana, Vladimir Atanasovski, Irene Macaluso, and Luiz A. DaSilva. "Learning and Reasoning in Cognitive Radio Networks", IEEE Communications Surveys & Tutorials, 2013. Crossref	26 words — < 1%
116	Tan Yigitcanlar. "Technology and the City - Systems, applications and implications", Routledge, 2016 Publications	26 words — < 1%
117	ethesis.nitrkl.ac.in Internet	26 words — < 1%
118	napier-repository.worktribe.com Internet	26 words — < 1%
119	"The Advancing World of Applied Electromagnetics", Springer Science and Business	25 words — < 1%

Media LLC, 2024
Crossref

120 Md. Rakib Hassan, Gour C. Karmakar, Joarder Kamruzzaman, Bala Srinivasan. "Exclusive Use Spectrum Access Trading Models in Cognitive Radio Networks: A Survey", IEEE Communications Surveys & Tutorials, 2017 — 25 words — < 1%
Crossref

121 Pengfei Sun. "A Guidebook for 5GtoB and 6G Vision for Deep Convergence", Springer Science and Business Media LLC, 2023 — 25 words — < 1%
Crossref

122 Ming Jin, Guo Chen, Qinghua Guo, Jun Li, Hai Li. "Log Likelihood Ratio Test for Spectrum Sensing With Truncated Covariance Matrix", IEEE Internet of Things Journal, 2024 — 24 words — < 1%
Crossref

123 Narasimha Swamy S, Kota Solomon Raju. "An Empirical Study on System Level Aspects of Internet of Things (IoT)", IEEE Access, 2020 — 24 words — < 1%
Crossref

124 hftelecomm.arizona.edu — 23 words — < 1%
Internet

125 journal.ijresm.com — 23 words — < 1%
Internet

126 researchbank.swinburne.edu.au — 23 words — < 1%
Internet

127 V. Sharmila, S. Kannadhasan, A. Rajiv Kannan, P. Sivakumar, V. Vennila. "Challenges in Information, Communication and Computing Technology", CRC Press, 2024 — 22 words — < 1%
Publications

128	paperhost.org Internet	22 words — < 1%
129	www.diplom.de Internet	22 words — < 1%
130	www.journal.mtu.edu.iq Internet	22 words — < 1%
131	Bin Zhou, Qishen Zhou, Simon Hu, Dongfang Ma, Sheng Jin, Der-Horng Lee. "Cooperative Traffic Signal Control Using a Distributed Agent-Based Deep Reinforcement Learning With Incentive Communication", IEEE Transactions on Intelligent Transportation Systems, 2024 Crossref	21 words — < 1%
132	Nidhi Sharma, Krishan Kumar. "Resource allocation trends for ultra dense networks in 5G and beyond networks: A classification and comprehensive survey", Physical Communication, 2021 Crossref	21 words — < 1%
133	Priya Mishra, Nisha Chaurasia. "Let's Identify the Similar and Confusing Raags of Hindustani Classical Music", 2023 3rd Asian Conference on Innovation in Technology (ASIANCON), 2023 Crossref	21 words — < 1%
134	eprints.lancs.ac.uk Internet	21 words — < 1%
135	Amol Dattatray Dhaygude, Suman Kumar Swarnkar, Priya Chugh, Yogesh Kumar Rathore. "Smart Agriculture - Harnessing Machine Learning for Crop Management", CRC Press, 2024 Publications	20 words — < 1%

136 Ashwin Rao, Tikhon Jelvis. "Foundations of Reinforcement Learning with Applications in Finance", CRC Press, 2022
Publications
20 words — < 1%

137 Ruhul Amin Khalil, Nasir Saeed, Mudassir Masood, Yasaman Moradi Fard, Mohamed-Slim Alouini, Tareq Y. Al-Naffouri. "Deep Learning in the Industrial Internet of Things: Potentials, Challenges, and Emerging Applications", IEEE Internet of Things Journal, 2021
Crossref
20 words — < 1%

138 Yunhao Liu, Zhihong Liu, Guanzheng Wang, Chao Yan, Xiangke Wang, Zhiping Huang. "Flexible multi-UAV formation control via integrating deep reinforcement learning and affine transformations", Aerospace Science and Technology, 2025
Crossref
20 words — < 1%

139 egrove.olemiss.edu
Internet
20 words — < 1%

140 iris.univpm.it
Internet
20 words — < 1%

141 jmlr.org
Internet
20 words — < 1%

142 xml.jips-k.org
Internet
20 words — < 1%

143 Tragos, Elias Z., Sherali Zeadally, Alexandros G. Fragkiadakis, and Vasilios A. Siris. "Spectrum Assignment in Cognitive Radio Networks: A Comprehensive Survey", IEEE Communications Surveys & Tutorials, 2013.
Crossref
19 words — < 1%

144 dspace.lib.uom.gr
Internet
19 words — < 1%

145 ict-solder.eu
Internet
19 words — < 1%

146 mmcalumni.ca
Internet
19 words — < 1%

147 Ahmed Khattab, Magdy A. Bayoumi. "An overview of IEEE standardization efforts for cognitive radio networks", 2015 IEEE International Symposium on Circuits and Systems (ISCAS), 2015
Crossref
18 words — < 1%

148 Arvind Dagur, Karan Singh, Pawan Singh Mehra, Dhirendra Kumar Shukla. "Artificial Intelligence, Blockchain, Computing and Security", CRC Press, 2023
Publications
18 words — < 1%

149 Arwa Aldweesh, Abdelouahid Derhab, Ahmed Z. Emam. "Deep learning approaches for anomaly-based intrusion detection systems: A survey, taxonomy, and open issues", Knowledge-Based Systems, 2020
Crossref
18 words — < 1%

150 Bhasker Dappuri, Madhu Babu Sikha, T.G. Venkatesh, Pamarthi Uma Sindhu. "An analytical framework for cooperative cognitive radio network for spectrum leasing/sharing", Physical Communication, 2024
Crossref
18 words — < 1%

151 Natasa Kleanthous, Abir Hussain. "Machine Learning in Farm Animal Behavior using Python", CRC Press, 2025
Publications
18 words — < 1%

152	Seowoo Jang, Jinguk Jeong, Juho Lee, Sunghyun Choi. "Digital Twin for Intelligent Network: Data Lifecycle, Digital Replication, and AI-based Optimizations", IEEE Communications Magazine, 2023 Crossref	18 words — < 1%
153	bmcgenomics.biomedcentral.com Internet	18 words — < 1%
154	jcis.sbrt.org.br Internet	18 words — < 1%
155	www.androcs.com Internet	18 words — < 1%
156	www.geeksforgeeks.org Internet	18 words — < 1%
157	Abdullah Hussein Talib, Aseel Hameed AL-Nakkash, Ahmed Ghanim Wadday. "A centralized cooperative study of spectrum sensing based on energy detection in cognitive radio networks", AIP Publishing, 2024 Crossref	17 words — < 1%
158	Adegoke Ayuba, Lawrence Kelvin, Terry Tera. "Quantum Computing and Machine Learning: Bridging the Gap for Future Innovations", Open Science Framework, 2023 Publications	17 words — < 1%
159	Jiang You, Zixun Ye, Jingliang Gu, Juntao Pu. "UAV-Pose: A Dual Capture Network Algorithm for Low Altitude UAV Attitude Detection and Tracking", IEEE Access, 2023 Crossref	17 words — < 1%

160	Mohamed Abdel-Basset, Nour Moustafa, Hossam Hawash, Weiping Ding. "Deep Learning Techniques for IoT Security and Privacy", Springer Science and Business Media LLC, 2022 Crossref	17 words — < 1%
161	Ton Duc Thang University Publications	17 words — < 1%
162	Xiang Tan, Li Zhou, Haijun Wang, Yuli Sun, Haitao Zhao, Boon-Chong Seet, Jibo Wei, Victor C. M. Leung. "Cooperative Multi-Agent Reinforcement-Learning-Based Distributed Dynamic Spectrum Access in Cognitive Radio Networks", IEEE Internet of Things Journal, 2022 Crossref	17 words — < 1%
163	ijser.aliraqia.edu.iq Internet	17 words — < 1%
164	palgronsund.com Internet	17 words — < 1%
165	scholar.uwindsor.ca Internet	17 words — < 1%
166	www.cai.sk Internet	17 words — < 1%
167	www.pure.ed.ac.uk Internet	17 words — < 1%
168	www.springerprofessional.de Internet	17 words — < 1%
169	G. Sucharitha, Anjanna Matta, M. Srinivas, Sachi Nandan Mohanty. "Artificial Intelligence Technologies for Engineering Applications", CRC Press, 2025	16 words — < 1%

Publications

170	krc.coe.neu.edu Internet	16 words — < 1%
171	singidunum.ac.rs Internet	16 words — < 1%
172	Eilif Hjelseth, Sujesh F. Sujan, Raimar J. Scherer. "ECPPM 2022 – eWork and eBusiness in Architecture, Engineering and Construction", CRC Press, 2023 Publications	15 words — < 1%
173	Felix Obite, Aliyu D. Usman, Emmanuel Okafor. "An overview of deep reinforcement learning for spectrum sensing in cognitive radio networks", Digital Signal Processing, 2021 Crossref	15 words — < 1%
174	Gioacchino Tangari, Muhammad Ikram, I Wayan Budi Sentana, Kiran Ijaz, Mohamed Ali Kaafar, Shlomo Berkovsky. "Analyzing security issues of android mobile health and medical applications", Journal of the American Medical Informatics Association, 2021 Crossref	15 words — < 1%
175	Zoran S. Bojkovic, Dragorad A. Milovanovic, Tulsi Pawan Fowdur. "5G Multimedia Communication - Technology, Multiservices, and Deployment", CRC Press, 2020 Publications	15 words — < 1%
176	patents.google.com Internet	15 words — < 1%
177	"Machine Learning for Future Wireless Communications", Wiley, 2020 Crossref	14 words — < 1%

178 123dok.net
Internet
14 words — < 1%

179 H.L. Gururaj, Francesco Flammini, S. Srividhya, M.L. Chayadevi, Sheba Selvam. "Computer Science Engineering", CRC Press, 2024
Publications
14 words — < 1%

180 Husheng Li, Robert C. Qiu. "A Graphical Framework for Spectrum Modeling and Decision Making in Cognitive Radio Networks", 2010 IEEE Global Telecommunications Conference GLOBECOM 2010, 2010
Crossref
14 words — < 1%

181 Jithin Jagannath, Nicholas Polosky, Anu Jagannath, Francesco Restuccia, Tommaso Melodia. "Machine learning for wireless communications in the Internet of Things: A comprehensive survey", Ad Hoc Networks, 2019
Crossref
14 words — < 1%

182 Qian Mao, Fei Hu, Qi Hao. "Deep Learning for Intelligent Wireless Networks: A Comprehensive Survey", IEEE Communications Surveys & Tutorials, 2018
Crossref
14 words — < 1%

183 Signals and Communication Technology, 2015.
Crossref
14 words — < 1%

184 Yih-Chun Hu, D.B. Johnson. "Design and demonstration of live audio and video over multihop wireless ad hoc networks", MILCOM 2002. Proceedings, 2002
Crossref
14 words — < 1%

185 Yuhua Xu, Zhan Gao, Wei Tian. "Optimal channel selection for cooperative spectrum sensing using
14 words — < 1%

coordination game", 7th International Conference on Communications and Networking in China, 2012
Crossref

186 medium.com
Internet
14 words — < 1%

187 www.momentslog.com
Internet
14 words — < 1%

188 "Handbook of Reinforcement Learning and Control", Springer Science and Business Media LLC, 2021
Crossref
13 words — < 1%

189 Ang Gao, Chengyuan Du, Soon Xin Ng, Wei Liang. "A Cooperative Spectrum Sensing with Multi-Agent Reinforcement Learning Approach in Cognitive Radio Networks", IEEE Communications Letters, 2021
Crossref
13 words — < 1%

190 Arjuna Madanayake, Keththura Lawrance, Bopage Umesha Kumarasiri, Sivakumar Sivasankar et al. "Design of Multichannel Spectrum Intelligence Systems Using Approximate Discrete Fourier Transform Algorithm for Antenna Array-Based Spectrum Perception Applications", Algorithms, 2024
Crossref
13 words — < 1%

191 Francisco Javier Campos Zabala. "Chapter 9 Supervised and Unsupervised Learning", Springer Science and Business Media LLC, 2023
Crossref
13 words — < 1%

192 J. Michael Spector, M. David Merrill, Jeroen van Merriënboer, Marcy P. Driscoll. "Handbook of
13 words — < 1%

Research on Educational Communications and Technology", Lawrence Erlbaum Associates, 2008
Publications

193 Md. Maruf Hossain Shuvo, Syed Kamrul Islam, Jianlin Cheng, Bashir I. Morshed. "Efficient Acceleration of Deep Learning Inference on Resource-Constrained Edge Devices: A Review", Proceedings of the IEEE, 2022 — 13 words — < 1%
Crossref

194 Seema Bhalgaonkar, Mousami Munot, Alwin anuse. "Model compression of deep neural network architectures for visual pattern recognition: Current status and future directions", Computers and Electrical Engineering, 2024 — 13 words — < 1%
Crossref

195 Vinicius F. Silva, Daniel F. Macedo, Jesse L. Leoni. "Spectrum Decision in Wireless Sensor Networks Employing Machine Learning", 2014 Brazilian Symposium on Computer Networks and Distributed Systems, 2014 — 13 words — < 1%
Crossref

196 english.dbw.cn — 13 words — < 1%
Internet

197 ieeexplore.ieee.org — 13 words — < 1%
Internet

198 internationalpubls.com — 13 words — < 1%
Internet

199 ndl.ethernet.edu.et — 13 words — < 1%
Internet

200 vtechworks.lib.vt.edu
Internet

13 words — < 1%

201 www.wiseguyreports.com
Internet
13 words — < 1%

202 Dinh C. Nguyen, Peng Cheng, Ming Ding, David Lopez-Perez et al. "Enabling AI in Future Wireless Networks: A Data Life Cycle Perspective", IEEE Communications Surveys & Tutorials, 2020
Crossref
12 words — < 1%

203 Himanshu Sharma, Neeraj Kumar. "Deep learning based physical layer security for terrestrial communications in 5G and beyond networks: A survey", Physical Communication, 2023
Crossref
12 words — < 1%

204 Kaylani Bochie, Mateus S. Gilbert, Luana Gantert, Mariana S.M. Barbosa, Dianne S.V. Medeiros, Miguel Elias M. Campista. "A survey on deep learning for challenged networks: Applications and trends", Journal of Network and Computer Applications, 2021
Crossref
12 words — < 1%

205 linnk.ai
Internet
12 words — < 1%

206 www.coursehero.com
Internet
12 words — < 1%

207 M. Geist, O. Pietquin. "Algorithmic Survey of Parametric Value Function Approximation", IEEE Transactions on Neural Networks and Learning Systems, 2013
Crossref
11 words — < 1%

208 Nasser Kimbugwe, Tingrui Pei, Moses Kyebambe. "Application of Deep Learning for Quality of Service Enhancement in Internet of Things: A Review", Energies, 2021
Crossref
11 words — < 1%

209 Tianlun Hu, Qi Liao, Qiang Liu, Georg Carle. "Network Slicing via Transfer Learning aided Distributed Deep Reinforcement Learning", GLOBECOM 2022 - 2022 IEEE Global Communications Conference, 2022
Crossref
11 words — < 1%

210 eprints.soton.ac.uk
Internet
11 words — < 1%

211 faculty.engineering.asu.edu
Internet
11 words — < 1%

212 www.ece.ucf.edu
Internet
11 words — < 1%

213 "The 10th International Conference on Computer Engineering and Networks", Springer Science and Business Media LLC, 2021
Crossref
10 words — < 1%

214 Anupama Namburu, Soubhagya Sankar Barpanda. "Recent Advances in Computer Based Systems, Processes and Applications", CRC Press, 2020
Publications
10 words — < 1%

215 Kai Hu, Mingyang Li, Zhiqiang Song, Keer Xu, Qingfeng Xia, Ning Sun, Peng Zhou, Min Xia. "A review of research on reinforcement learning algorithms for multi-agents", Neurocomputing, 2024
Crossref
10 words — < 1%

216	Laxminarayana S Pillutla, Bhuvan Joshi. "Sequential cooperative spectrum sensing in cognitive radio networks: Optimal selection of secondary users and their spectral measurements", 2017 9th International Conference on Communication Systems and Networks (COMSNETS), 2017 Crossref	10 words — < 1%
217	Marketing Intelligence & Planning, Volume 26, Issue 2 (2008-04-06) Publications	10 words — < 1%
218	Nahid Parvez Farazi, Bo Zou, Tanvir Ahamed, Limon Barua. "Deep reinforcement learning in transportation research: A review", Transportation Research Interdisciplinary Perspectives, 2021 Crossref	10 words — < 1%
219	Sujata Dash, Subhendu Kumar Pani, Joel J. P. C. Rodrigues, Babita Majhi. "Deep Learning, Machine Learning and IoT in Biomedical and Health Informatics - Techniques and Applications", CRC Press, 2022 Publications	10 words — < 1%
220	Ying Luo, Min Zeng, Hong Jiang. "Learning to Tradeoff Between Energy Efficiency and Delay in Energy Harvesting-Powered D2D Communication: A Distributed Experience-Sharing Algorithm", IEEE Internet of Things Journal, 2019 Crossref	10 words — < 1%
221	assets-eu.researchsquare.com Internet	10 words — < 1%
222	eprints.utar.edu.my Internet	10 words — < 1%

223 "Advances in Artificial Intelligence", Springer Science and Business Media LLC, 2017 — 9 words — < 1%
Crossref

224 "Advances in Machine Learning and Computational Intelligence", Springer Science and Business Media LLC, 2021 — 9 words — < 1%
Crossref

225 "Artificial Neural Networks and Machine Learning – ICANN 2017", Springer Science and Business Media LLC, 2017 — 9 words — < 1%
Crossref

226 "Chapter 284 The Orange Model", Springer Science and Business Media LLC, 2008 — 9 words — < 1%
Crossref

227 "Intelligence Science and Big Data Engineering. Visual Data Engineering", Springer Science and Business Media LLC, 2019 — 9 words — < 1%
Crossref

228 Chen Zhong, Ziyang Lu, M. Cenk Gursoy, Senem Velipasalar. "A Deep Actor-Critic Reinforcement Learning Framework for Dynamic Multichannel Access", IEEE Transactions on Cognitive Communications and Networking, 2019 — 9 words — < 1%
Crossref

229 Fei Hu. "UAV Swarm Networks - Models, Protocols, and Systems", CRC Press, 2020 — 9 words — < 1%
Publications

230 Gebrekiros Gebreyesus Gebremariam, J. Panda, S. Indu. "Secure localization techniques in wireless sensor networks against routing attacks based on hybrid — 9 words — < 1%

machine learning models", Alexandria Engineering Journal, 2023
Crossref

231 Gonçalo Hora de Carvalho, Twan Vos. "Deep Reinforcement Learning in High-Dimensional and Stochastic Environments A Short Introductory Literature Survey & Critical Methodological Review", Open Science Framework, 2024 — 9 words — < 1%
Publications

232 Jose M. de la Rosa. "AI-Managed Cognitive Radio Digitizers", IEEE Circuits and Systems Magazine, 2022 — 9 words — < 1%
Crossref

233 Kok-Lim Alvin Yau, Geong-Sen Poh, Su Fong Chien, Hasan A. A. Al-Rawi. "Application of Reinforcement Learning in Cognitive Radio Networks: Models and Algorithms", The Scientific World Journal, 2014 — 9 words — < 1%
Crossref

234 Marko Hoyhtya, Aarne Mammela, Marina Eskola, Marja Matinmikko et al. "Spectrum Occupancy Measurements: A Survey and Use of Interference Maps", IEEE Communications Surveys & Tutorials, 2016 — 9 words — < 1%
Crossref

235 Ola Salman, Imad H. Elhajj, Ali Chehab, Ayman Kayssi. "A Multi-level Internet Traffic Classifier Using Deep Learning", 2018 9th International Conference on the Network of the Future (NOF), 2018 — 9 words — < 1%
Crossref

236 Ramjee Prasad. "Future Trends and Challenges for ICT Standardization", Routledge, 2025 — 9 words — < 1%
Publications

237	Saad Z. Asif. " 5G Mobile Communications Concepts and Technologies", CRC Press, 2018 Publications	9 words — < 1%
238	Saravanan Krishnan, A. Jose Anand, S. Sendhilkumar. "Handbook of Industrial and Business Applications with Digital Twins", Routledge, 2024 Publications	9 words — < 1%
239	Srdjan S. Stankovic, Marko Beko, Milos S. Stankovic. "Robust Nonlinear Consensus Seeking", 2019 IEEE 58th Conference on Decision and Control (CDC), 2019 Crossref	9 words — < 1%
240	Tianheng Song, Dazi Li, Qibing Jin, Kotaro Hirasawa. "Sparse Proximal Reinforcement Learning via Nested Optimization", IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2018 Crossref	9 words — < 1%
241	Wenzelburger, J.. "Learning in linear models with expectational leads", Journal of Mathematical Economics, 200611 Crossref	9 words — < 1%
242	Wu, Cheng, Y.M. Wang, Xiang Qiang, and Z.Y. Zhang. "Adaptive Spectrum Management of Cognitive Radio in Intelligent Transportation System", Applied Mechanics and Materials, 2015. Crossref	9 words — < 1%
243	depot.umc.edu.dz Internet	9 words — < 1%
244	dergipark.org.tr Internet	9 words — < 1%

245	di.univ-blida.dz Internet	9 words — < 1%
246	jultika.oulu.fi Internet	9 words — < 1%
247	kclpure.kcl.ac.uk Internet	9 words — < 1%
248	prr.hec.gov.pk Internet	9 words — < 1%
249	pure.tue.nl Internet	9 words — < 1%
250	repository.up.ac.za:8080 Internet	9 words — < 1%
251	www.irejournals.com Internet	9 words — < 1%
252	www.isy.liu.se Internet	9 words — < 1%
253	"Cognitive Radio, Mobile Communications and Wireless Networks", Springer Science and Business Media LLC, 2019 Crossref	8 words — < 1%
254	"Machine Intelligence and Big Data Analytics for Cybersecurity Applications", Springer Science and Business Media LLC, 2021 Crossref	8 words — < 1%
255	Adeeb Salh, Lukman Audah, Nor Shahida Mohd Shah, Abdulraqeb Alhammadi et al. "A Survey on Deep Learning for Ultra-Reliable and Low-Latency	8 words — < 1%

Communications Challenges on 6G Wireless Systems", IEEE Access, 2021
Crossref

256 Ahmed Al-Tahmeesschi, Kenta Umebayashi, Hiroki Iwata, Janne Lehtomaki, Miguel Lopez-Benitez. "Feature-Based Deep Neural Networks for Short-Term Prediction of WiFi Channel Occupancy Rate", IEEE Access, 2021 — 8 words — < 1%
Crossref

257 Bkassiny, Mario, Yang Li, and Sudharman K. Jayaweera. "A Survey on Machine-Learning Techniques in Cognitive Radios", IEEE Communications Surveys & Tutorials, 2012. — 8 words — < 1%
Crossref

258 Caiwen Wang, Binghan Zheng. "Empirical Studies of Translation and Interpreting - The Post-Structuralist Approach", Routledge, 2021 — 8 words — < 1%
Publications

259 Ejaz, Waleed, Najam ul Hasan, and Hyung Seok Kim. "Distributed cooperative spectrum sensing in cognitive radio for ad hoc networks", Computer Communications, 2013. — 8 words — < 1%
Crossref

260 Fadi Al-Turjman. "Smart Grid in IoT-Enabled Spaces - The Road to Intelligence in Power", CRC Press, 2020 — 8 words — < 1%
Publications

261 Fei Hu. "Big Data - Storage, Sharing, and Security", Auerbach Publications, 2019 — 8 words — < 1%
Publications

262 Francesco Musumeci, Cristina Rottondi, Avishek Nag, Irene Macaluso, Darko Zibar, Marco Ruffini, — 8 words — < 1%

Massimo Tornatore. "An Overview on Application of Machine Learning Techniques in Optical Networks", IEEE Communications Surveys & Tutorials, 2019
Crossref

263 Gabriella Melki, Vojislav Kecman, Sebastián Ventura, Alberto Cano. "OLLAWV: OnLine Learning Algorithm using Worst-Violators", Applied Soft Computing, 2018 — 8 words — < 1%
Crossref

264 Gunjan Beniwal, Anita Singhrova. ""A systematic literature review on IoT gateways"", Journal of King Saud University - Computer and Information Sciences, 2022 — 8 words — < 1%
Crossref

265 Ishfaq Ahmad, Sanjay Ranka. "Handbook of Energy-Aware and Green Computing, Volume 2", Chapman and Hall/CRC, 2013 — 8 words — < 1%
Publications

266 Jeroen Graaf. "Anonymous One-Time Broadcast Using Non-interactive Dining Cryptographer Nets with Applications to Voting", Lecture Notes in Computer Science, 2010 — 8 words — < 1%
Crossref

267 Joseph D. Bronzino, Donald R. Peterson. "Biomedical Engineering Fundamentals", CRC Press, 2019 — 8 words — < 1%
Publications

268 Junqi Jin, Chengru Song, Han Li, Kun Gai, Jun Wang, Weinan Zhang. "Real-Time Bidding with Multi-Agent Reinforcement Learning in Display Advertising", Proceedings of the 27th ACM International Conference on Information and Knowledge Management - CIKM '18, 2018 — 8 words — < 1%
Crossref

269 Lecture Notes in Electrical Engineering, 2016.
Crossref
8 words — < 1%

270 Miroslav V. Peric, Dragana B. Peric, Branislav M. Todorovic, Miroslav V. Popovic. "Dynamic Rain Attenuation Model for Millimeter Wave Network Analysis", IEEE Transactions on Wireless Communications, 2017
Crossref
8 words — < 1%

271 Muhammad Aneeq uz Zaman, Kaiqing Zhang, Erik Miehling, Tamer Basar. "Approximate Equilibrium Computation for Discrete-Time Linear-Quadratic Mean-Field Games", 2020 American Control Conference (ACC), 2020
Crossref
8 words — < 1%

272 Payal Khurana Batra, Pawan Singh Mehra, Sudeep Tanwar. "Network Optimization in Intelligent Internet of Things Applications - Principles and Challenges", CRC Press, 2024
Publications
8 words — < 1%

273 Qusay F. Hassan, Rehman Khan Atta ur, Sajjad A. Madani. "Internet of Things - Challenges, Advances, and Applications", CRC Press, 2017
Publications
8 words — < 1%

274 Ren Han, Yang Gao, Chunxue Wu, Dianjie Lu. "An Effective Multi-Objective Optimization Algorithm for Spectrum Allocations in the Cognitive-Radio-Based Internet of Things", IEEE Access, 2018
Crossref
8 words — < 1%

275 Ruqiang Yan, Zhibin Zhao. "Deep Neural Networks-Enabled Intelligent Fault Diagnosis of Mechanical Systems", CRC Press, 2024
Publications
8 words — < 1%

276 S.J. Xavier Savarimuthu, Sivakannan Subramani, Alex Noel Joseph Raj. "Artificial Intelligence for Multimedia Information Processing - Tools and Applications", CRC Press, 2024 — 8 words — < 1%
Publications

277 Sepp Hochreiter, Jürgen Schmidhuber. "Long Short-Term Memory", Neural Computation, 1997 — 8 words — < 1%
Crossref

278 Shilian Zheng, Shichuan Chen, Lifeng Yang, Jiawei Zhu, Zhenxing Luo, Junjie Hu, Xiaoniu Yang. "Big Data Processing Architecture for Radio Signals Empowered by Deep Learning: Concept, Experiment, Applications and Challenges", IEEE Access, 2018 — 8 words — < 1%
Crossref

279 Susmita Bandyopadhyay, Ranjan Bhattacharya. "Discrete and Continuous Simulation - Theory and Practice", CRC Press, 2019 — 8 words — < 1%
Publications

280 Tapiwa M. Chiwewe, Colman F. Mbuya, Gerhard P. Hancke. "Using Cognitive Radio for Interference-Resistant Industrial Wireless Sensor Networks: An Overview", IEEE Transactions on Industrial Informatics, 2015 — 8 words — < 1%
Crossref

281 Tianjing Wang, Xinjie Guan, Xili Wan, Hang Shen, Xiaomei Zhu. "A Spectrum-Aware Clustering Algorithm Based on Weighted Clustering Metric in Cognitive Radio Sensor Networks", IEEE Access, 2019 — 8 words — < 1%
Crossref

282 Yaohua Sun, Mugen Peng, Yangcheng Zhou, Yuzhe Huang, Shiwen Mao. "Application of Machine — 8 words — < 1%

Learning in Wireless Networks: Key Techniques and Open Issues", IEEE Communications Surveys & Tutorials, 2019
Crossref

283 Ye Xiao, Xingchen Li, Wen Yao, Jin Chen, Yupeng Hu. "Bidirectional Data-Driven Trajectory Prediction for Intelligent Maritime Traffic", IEEE Transactions on Intelligent Transportation Systems, 2022 — 8 words — < 1%
Crossref

284 Zheng, S., C. Lou, and X. Yang. "Cooperative spectrum sensing using particle swarm optimisation", Electronics Letters, 2010. — 8 words — < 1%
Crossref

285 ar5iv.org — 8 words — < 1%
Internet

286 archive.org — 8 words — < 1%
Internet

287 backoffice.biblio.ugent.be — 8 words — < 1%
Internet

288 digi.lib.ttu.ee — 8 words — < 1%
Internet

289 dlibra.itl.waw.pl — 8 words — < 1%
Internet

290 hal.science — 8 words — < 1%
Internet

291 idus.us.es — 8 words — < 1%
Internet

292 ijeecs.iaescore.com — 8 words — < 1%
Internet

293	link.springer.com Internet	8 words — < 1%
294	manualzz.com Internet	8 words — < 1%
295	mascots20.iitis.pl Internet	8 words — < 1%
296	opus.lib.uts.edu.au Internet	8 words — < 1%
297	orbilu.uni.lu Internet	8 words — < 1%
298	oulurepo.oulu.fi Internet	8 words — < 1%
299	publications.rwth-aachen.de Internet	8 words — < 1%
300	repositorio.upct.es Internet	8 words — < 1%
301	scholar.colorado.edu Internet	8 words — < 1%
302	scholar.sun.ac.za Internet	8 words — < 1%
303	slideplayer.com Internet	8 words — < 1%
304	smartech.gatech.edu Internet	8 words — < 1%

305	studylib.net Internet	8 words — < 1%
306	swsys-web.ru Internet	8 words — < 1%
307	tel.archives-ouvertes.fr Internet	8 words — < 1%
308	theses.hal.science Internet	8 words — < 1%
309	vbn.aau.dk Internet	8 words — < 1%
310	www.fitee.zjujournals.com Internet	8 words — < 1%
311	www.ijiemr.org Internet	8 words — < 1%
312	www.imstat.org Internet	8 words — < 1%
313	www.kn-s.dlr.de Internet	8 words — < 1%
314	Amandeep Kaur, Krishan Kumar. "Intelligent spectrum management based on reinforcement learning schemes in cooperative cognitive radio networks", Physical Communication, 2020 Crossref	7 words — < 1%
315	Anna Maria Vegni, Dharma P. Agrawal. "Cognitive Vehicular Networks", CRC Press, 2018 Publications	7 words — < 1%

316 Giuseppe Caso, Ozgu Alay, Guido Carlo Ferrante, Luca De Nardis, Maria-Gabriella Di Benedetto, Anna Brunstrom. "User-Centric Radio Access Technology Selection: A Survey of Game Theory Models and Multi-Agent Learning Algorithms", IEEE Access, 2021
Crossref
7 words — < 1%

317 Habibul Islam, Ying-Chang Liang, Anh Tuan Hoang. "Joint Beamforming and Power Control in the Downlink of Cognitive Radio Networks", 2007 IEEE Wireless Communications and Networking Conference, 2007
Crossref
7 words — < 1%

318 Lecture Notes in Computer Science, 2015.
Crossref
7 words — < 1%

319 Manish Kumar Giri, Saikat Majumder. "Distributed dynamic spectrum access through multi-agent deep recurrent Q-learning in cognitive radio network", Physical Communication, 2023
Crossref
7 words — < 1%

320 Nadia Kassri, Abdeslam Ennouaary, Slimane Bah, Hajar Baghdadi. "A Review on SDR, Spectrum Sensing, and CR-based IoT in Cognitive Radio Networks", International Journal of Advanced Computer Science and Applications, 2021
Crossref
7 words — < 1%

321 Robert C. Qiu, Zhen Hu, Husheng Li, Michael C. Wicks. "Cognitive Radio Communications and Networking", Wiley, 2012
Crossref
7 words — < 1%

322 Shweta Pandit, Ghanshyam Singh. "Spectrum Sharing in Cognitive Radio Networks", Springer Science and Business Media LLC, 2017
7 words — < 1%

Crossref

323 Yan Zhang, Hsiao-Hwa Chen, Mohsen Guizani. "Cooperative Wireless Communications", Auerbach Publications, 2019 — 7 words — < 1%
Publications

324 Yuhua Xu, Jinlong Wang, Qihui Wu, Jianchao Zheng, Liang Shen, Alagan Anpalagan. "Dynamic Spectrum Access in Time-Varying Environment: Distributed Learning Beyond Expectation Optimization", IEEE Transactions on Communications, 2017 — 7 words — < 1%
Crossref

325 "6G Mobile Wireless Networks", Springer Science and Business Media LLC, 2021 — 6 words — < 1%
Crossref

326 "Collaborative Computing: Networking, Applications and Worksharing", Springer Nature, 2019 — 6 words — < 1%
Crossref

327 "Information Security Applications", Springer Nature, 2017 — 6 words — < 1%
Crossref

328 "White Space Communication", Springer Science and Business Media LLC, 2015 — 6 words — < 1%
Crossref

329 Ashish Bagwari, Geetam Singh Tomar, Jyotshana Bagwari. " Advanced Wireless Sensing Techniques for 5G Networks", CRC Press, 2018 — 6 words — < 1%
Publications

330 Chapman Siu. "Residual Networks Behave Like Boosting Algorithms", 2019 IEEE International — 6 words — < 1%

Conference on Data Science and Advanced Analytics (DSAA), 2019
Crossref

331 Charu C. Aggarwal. "Neural Networks and Deep Learning", Springer Science and Business Media LLC, 2018 — 6 words — < 1%
Crossref

332 Dan Hendrycks. "Introduction to AI Safety, Ethics, and Society", CRC Press, 2024 — 6 words — < 1%
Publications

333 Donghwan Lee, Niao He, Parameswaran Kamalaruban, Volkan Cevher. "Optimization for Reinforcement Learning: From a single agent to cooperative agents", IEEE Signal Processing Magazine, 2020 — 6 words — < 1%
Crossref

334 Fei Hu, Iftikhar Rasheed. "Deep Learning and Its Applications for Vehicle Networks", CRC Press, 2023 — 6 words — < 1%
Publications

335 Geoffrey E. Hinton, Simon Osindero, Yee-Whye Teh. "A Fast Learning Algorithm for Deep Belief Nets", Neural Computation, 2006 — 6 words — < 1%
Crossref

336 Huynh Thanh Thien, Van-Hiep Vu, Insoo Koo. "A Transfer Games Actor–Critic Learning Framework for Anti-Jamming in Multi-Channel Cognitive Radio Networks", IEEE Access, 2021 — 6 words — < 1%
Crossref

337 Inam Ullah, Inam Ullah Khan, Mariya Ouaissa, Mariyam Ouaissa, Salma El Hajjami. "Future — 6 words — < 1%

Communication Systems Using Artificial Intelligence, Internet of Things and Data Science", CRC Press, 2024
Publications

338 John D. Matyjas, Sunil Kumar, Fei Hu. "Spectrum Sharing in Wireless Networks - Fairness, Efficiency, and Security", CRC Press, 2019 — 6 words — < 1%
Publications

339 Khalid A. Darabkh, Oswa M. Amro, Haythem Bany Salameh, Raed T. Al-Zubi. "A–Z overview of the in-band full-duplex cognitive radio networks", Computer Communications, 2019 — 6 words — < 1%
Crossref

340 Mduduzi C. Hlophe, Bodhaswar T. Maharaj. "AI Meets CRNs: A Prospective Review on the Application of Deep Architectures in Spectrum Management", IEEE Access, 2021 — 6 words — < 1%
Crossref

341 Qingguo Yang, Honghu Wang, Jiayan Zheng, Wencheng Cheng, Shuangchen Xia. "Prediction of mechanical properties of basalt fiber concrete using hybrid recurrent neural networks based on freeze-thaw damage quantification", Journal of Building Engineering, 2024 — 6 words — < 1%
Crossref

342 Sizhuang Liu, Tengjiao Wang, Changyong Pan, Chao Zhang, Fang Yang, Jian Song. "Deep Reinforcement Learning for Spectrum Sharing in Future Mobile Communication System", 2021 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB), 2021 — 6 words — < 1%
Crossref

343 Tevfik Yucek, Huseyin Arslan. "Spectrum Characterization for Opportunistic Cognitive Radio — 6 words — < 1%

Systems", MILCOM 2006, 2006
Crossref

344 Warren B. Powell. "Perspectives of approximate dynamic programming", Annals of Operations Research, 2012 — 6 words — < 1%
Crossref

345 Wenbo Wang, Andres Kwasinski, Dusit Niyato, Zhu Han. "A Survey on Applications of Model-Free Strategy Learning in Cognitive Wireless Networks", IEEE Communications Surveys & Tutorials, 2016 — 6 words — < 1%
Crossref

346 Xiaofeng Yang. "Medical Image Synthesis - Methods and Clinical Applications", CRC Press, 2024 — 6 words — < 1%
Publications

347 Yuan Luo, Lin Gao, Jianwei Huang. "Economics of Database-Assisted Spectrum Sharing", Springer Science and Business Media LLC, 2016 — 6 words — < 1%
Crossref

EXCLUDE QUOTES	OFF	EXCLUDE SOURCES	OFF
EXCLUDE BIBLIOGRAPHY	OFF	EXCLUDE MATCHES	OFF