



**HAL**  
open science

# Development of Spectrum Sharing Protocol for Cognitive Radio Internet of Things

Dina Tarek Mohamed Ibrahim Hafez

► **To cite this version:**

Dina Tarek Mohamed Ibrahim Hafez. Development of Spectrum Sharing Protocol for Cognitive Radio Internet of Things. Other [cs.OH]. Université d'Avignon; American university in Cairo, 2020. English. NNT : 2020AVIG0280 . tel-03164966

**HAL Id: tel-03164966**

**<https://theses.hal.science/tel-03164966>**

Submitted on 10 Mar 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Cairo University  
Faculty of Computers and Artificial  
Intelligence  
Information Technology Department



# Development of Spectrum Sharing Protocol for Cognitive Radio Internet of Things

*By*

**Dina Tarek Mohamed Ibrahim Hafez**

A Thesis Submitted to the  
Faculty of Computers and Artificial Intelligence - Cairo University  
and  
University of Avignon  
In Partial Fulfillment of the  
Requirements for the Dual Philosophy Degree  
In Information Technology

*Under supervision*

**Prof. Dr. Mohamed Gamal Eldin Darwish**

**Prof. Dr. Amira Mohamed Kotb**

**Cairo University, Egypt**

**Prof. Dr. Abderrahim Benslimane**

**University of Avignon, France**

**2020**



**Cairo University**  
**Faculty of Computers and Artificial**  
**Intelligence**  
**Information Technology Department**



UNIVERSITÉ D'AVIGNON  
ET DES PAYS DE VAUCLUSE

# **Development of Spectrum Sharing Protocol for Cognitive Radio Internet of Things**

*By*

**Dina Tarek Mohamed Ibrahim Hafez**

A Thesis Submitted to the  
Faculty of Computers and Artificial Intelligence - Cairo University

and

University of Avignon

In Partial Fulfillment of the  
Requirements for the Dual Philosophy Degree  
In Information Technology

Approved by the Examiner Committee

# Declaration

I certify that this work has not been accepted in substance for any academic degree and is not being concurrently submitted in candidature for any other degree. Any portion of this thesis for which I am indebted to other sources are mentioned and explicit references are given.

Student Name: Dina Tarek Mohamed Ibrahim Hafez

Signature:

# Dedication

This thesis is dedicated to my mother Imane Aly Saroit and father Tarek Mohamed Ibrahim, without their love and care it would have been impossible to accomplish this dream. My Husband Ahmed Mohamed Sarwat without his care and dedication I wouldn't be able finish this thesis at that time. And to my adorable daughter Lily, you have made me stronger, better and more fulfilled than I could have ever imagined. I love you to the moon and back. To my lovely older brother Mohamed Tarek Mohamed without his constant support it wouldn't have been possible to be successful in my life.

# Acknowledgment

All my praise to God my source of wisdom, knowledge and power, his grace provided me with the patience and power to accomplish this work.

All my thanks would be to my row model my mother **Professor. Imane Aly Saroit** without whom I wouldn't be able to accomplish anything in my life; I thank her for giving me the power to accomplish my dreams, and making me believe in myself. She put me on the road to become a good teacher and researcher. She taught me to be dedicated to my work, my research and my faculty. She taught me that if I have any work I should do it perfectly. She is my life, I thank God making her my mother, without her love it wouldn't have been possible for me to accomplish my dreams. I own her for being where I am.

Also I would like to thanks my father **Tarek Hafez** without his prayers and constant support it would not have been possible to travel or finish this work, thanks for standing by me in my whole life, trusting all of my choices; no word can express my appreciation and thanks to him.

I would like to express my sincere gratitude to my French supervisor **Professor. Abderrahim Benslimane** for the continuous support of my Ph. D studies and related research, for his patience and motivation, also for providing me with his valuable time and academic knowledge. His guidance helped me in my research and writing of this thesis. Then, I would like to express my deep gratitude to my Egyptian supervisors **Professor. Mohamed Gamal El-Din Darwish** and **Professor. Amira Mohamed Kotb**. I thank them for their constant help by providing me their precious time, and their academic and professional advices through the entire thesis, and also for their guidance throughout my studies. I share the credit of this work with my supervisors.

Special thanks for my daughter **Lily Sarwat** for being good kid with your mother when I was studying and bearing our travels, and my husband **Ahmed**

**Sarwat** without his caring supports it would not have been possible to finish this work, thanks for encouraging me, supporting me and being patient with me till I finished this thesis. Thanks to my older brother **Mohamed Hafez** and the rest of my family for all of the love, support, encouragement and prayers they have sent my way along this journey and for making me believe in myself and making me trust that I could do better.

Some special words of gratitude go to my friend and my struggle mate **Randa Mohamed Abdel Monem**; we had several ups and downs however you have always been a major source of support when things would get a bit discouraging.

Also I would like to thank my friends **Dr. Eman Ahmed, Heba Sheif** and **Dr. Shaimaa Mussad** for their encouragement for supporting me and giving me their experience, may thanks to you.

And finally, I would like to express my special appreciation for my faculty honorable teachers **Professor. Mahmoud Shoman** and **Professor. Reda El-Khoribi** for their academic and administrative guidance.

# Table of Contents

<b>Declaration</b> .....	<b>i</b>
<b>Dedication</b> .....	<b>ii</b>
<b>Acknowledgment</b> .....	<b>iii</b>
<b>Table of Contents</b> .....	<b>v</b>
<b>List of Figures</b> .....	<b>ix</b>
<b>List of Tables</b> .....	<b>xi</b>
<b>List of Abbreviations</b> .....	<b>xii</b>
<b>Abstract</b> .....	<b>xiv</b>
<b>Abstrait</b> .....	<b>xvi</b>
<b>Chapter 1</b> .....	<b>2</b>
<b>Introduction</b> .....	<b>2</b>
<b>1.1 Problem Description</b> .....	<b>2</b>
1.1.1 Internet of Things .....	2
1.1.2 Cognitive Radio Networks .....	3
1.1.3 Cognitive Radio Internet of Things.....	3
<b>1.2 Thesis Organization</b> .....	<b>4</b>
<b>Chapter 2</b> .....	<b>8</b>
<b>Cognitive Radio Networks</b> .....	<b>8</b>
<b>2.1 Cognitive Radio and Cognitive Radio Networks</b> .....	<b>8</b>
2.1.1 Cognitive Radio History.....	10
2.1.2 Cognitive Radio Concept and Abilities.....	11
2.1.3 Cognitive Radio Network Definition .....	12
<b>2.2 Cognitive Radio Networks Elements</b> .....	<b>12</b>
2.2.1 Primary Network .....	13
2.2.2 Cognitive Radio Network.....	14
<b>2.3 Cognitive Radio Network Classification</b> .....	<b>15</b>
2.3.1 Centralized (Infrastructure) Architecture .....	15
2.3.2 Distributed (Ad hoc) Architecture.....	16
2.3.3 Hybrid (Mesh) Architecture .....	16
<b>2.4 Cognitive Radio Main Functions</b> .....	<b>17</b>



2.4.1	Spectrum Sensing.....	18
2.4.2	Spectrum Management.....	21
2.4.3	Spectrum Mobility.....	22
2.4.4	Spectrum Sharing .....	23
<b>2.5</b>	<b>Cognitive Radio Layer Design.....</b>	<b>28</b>
2.5.1	Spectrum Sensing Functions' Involved Layer.....	28
2.5.2	Spectrum Sharing Functions' Involved Layer .....	28
2.5.3	Spectrum Management Functions' Involved Layers .....	29
2.5.4	Spectrum Mobility Functions' Involved Layers .....	30
<b>2.6</b>	<b>Cognitive Radio Networks Applications.....</b>	<b>30</b>
<b>2.7</b>	<b>Cognitive Radio Networks Advantages and Disadvantages ....</b>	<b>32</b>
2.7.1	Cognitive Radio Networks' Advantages .....	32
2.7.2	Cognitive Radio Disadvantages .....	32
<b>2.8</b>	<b>Summary .....</b>	<b>33</b>
<b>Chapter 3</b>	<b>.....</b>	<b>35</b>
<b>Internet of Things.....</b>	<b>.....</b>	<b>35</b>
<b>3.1 IoT Definition .....</b>	<b>.....</b>	<b>35</b>
<b>3.2 Internet of Things Architecture .....</b>	<b>.....</b>	<b>37</b>
3.2.1	Three Layered Architecture .....	38
3.2.2	Four Layered Architecture .....	38
3.2.3	Five Layered Architecture.....	38
<b>3.3 Internet of Things Communication Technologies .....</b>	<b>.....</b>	<b>39</b>
3.3.1	Short Range Communication Technologies.....	39
3.3.2	Long Range Communication Technologies .....	40
<b>3.4 Internet of Things Applications.....</b>	<b>.....</b>	<b>41</b>
3.4.1	Smart Home and Offices Applications.....	42
3.4.2	Smart City .....	42
3.4.3	Transportation Applications.....	43
3.4.4	Healthcare.....	44
3.4.5	Smart Grid .....	45
3.4.6	Smart Agriculture.....	46
<b>3.5 Fog Computing .....</b>	<b>.....</b>	<b>47</b>
<b>3.6 Summary .....</b>	<b>.....</b>	<b>49</b>
<b>Chapter 4 .....</b>	<b>.....</b>	<b>51</b>
<b>Spectrum Sharing/Allocation for.....</b>	<b>.....</b>	<b>51</b>
<b>Cognitive Radio Networks Internet of .....</b>	<b>.....</b>	<b>51</b>

<b>Things.....</b>	<b>51</b>
<b>4.1 Cognitive Radio Internet of Things (CRIoTs).....</b>	<b>51</b>
<b>4.2 Cognitive Radio Internet of Things Challenges.....</b>	<b>54</b>
<b>4.3 Cognitive Radio Internet of Things Spectrum Assignment Survey. 55</b>	
4.3.1 Spectrum Sharing .....	56
4.3.2 Spectrum Allocation.....	59
<b>4.4 Conclusion .....</b>	<b>67</b>
<b>Chapter 5 .....</b>	<b>69</b>
<b>Cognitive Radio Networks Channel State Estimation Using Machine Learning</b>	
<b>Techniques .....</b>	<b>69</b>
<b>5.1 Related Work .....</b>	<b>70</b>
<b>5.2 The Proposed Protocols.....</b>	<b>71</b>
5.2.1 Prediction using Bayes' Theorem and HMM (2-Model HMM).....	73
5.2.2 Prediction using SVM with HMM parameters of the two models (2-Model HMM-SVM).....	76
<b>5.3 Protocols Evaluation.....</b>	<b>78</b>
5.3.1 Simulation Parameters.....	79
5.3.2 Performance Parameters.....	80
5.3.3 Simulation Results.....	81
5.3.4 Analysis and Drawbacks .....	87
<b>5.4 Conclusion .....</b>	<b>88</b>
<b>Chapter 6 .....</b>	<b>92</b>
<b>Centralized Packets Scheduling Technique for Cognitive Radio Internet of</b>	
<b>Things Based on Discrete Permutation Particle Swarm Optimization.....</b>	<b>92</b>
<b>6.1 Background .....</b>	<b>93</b>
6.1.1 Particle Swarm Optimization .....	93
6.1.2 Spectrum Auction Technique.....	96
<b>6.2 Problem Formulation .....</b>	<b>98</b>
6.2.1 Mathematical Formulation .....	98
6.2.2 Problem Explanation.....	100
<b>6.3 Proposed Protocol.....</b>	<b>103</b>
<b>6.4 Perfomance Evaluation .....</b>	<b>109</b>
6.4.1 Protocols' complexity .....	109
6.4.2 Simulation Evaluation.....	111
6.4.3 Results analysis .....	119

<b>6.5 Conclusion .....</b>	<b>121</b>
<b>Chapter 7 .....</b>	<b>123</b>
<b>Distributed Packets Scheduling Technique for Cognitive Radio Internet of Things Based on Discrete Permutation Particle Swarm Optimization.....</b>	<b>123</b>
<b>7.1 Distributed Scheduling.....</b>	<b>124</b>
<b>7.2 Concept of the Proposed Distributed Scheduling.....</b>	<b>125</b>
7.2.1 System Structure .....	126
7.2.2 Problem Formulation.....	128
7.2.3 Proposed Distributed Medium Access .....	132
<b>7.3 Performance Evaluation .....</b>	<b>137</b>
7.3.1 Performance Evaluation Metrics .....	138
7.3.2 Simulation Parameters.....	139
7.3.3 Simulation Results.....	140
7.3.4 Result Analysis.....	147
<b>7.4 Conclusion .....</b>	<b>149</b>
<b>Chapter 8 .....</b>	<b>152</b>
<b>Conclusions and Future Works .....</b>	<b>152</b>
<b>8.1 Conclusion .....</b>	<b>152</b>
<b>8.2 Future Works.....</b>	<b>154</b>
<b>References .....</b>	<b>157</b>
<b>Outcome from the Thesis .....</b>	<b>173</b>

# List of Figures

Figure 2.1 Radio Spectrum Holes Concept .....	10
Figure 2.2 Cognitive Radio Network Architecture .....	13
Figure 2.3 Infrastructure Architecture .....	15
Figure 2.4 Ad-Hoc Architecture .....	16
Figure 2.5 Mesh Architecture .....	17
Figure 2.6 Cognitive Radio Cycle .....	18
Figure 2.7 Transmitter Detection Problems .....	20
Figure 2.8 Interference Temperature Model Spectrum Management .....	21
Figure 2.9 Classifications of Medium Access Control Protocols in Cognitive Radio Networks .....	24
Figure 2.10 Cooperative and Non-Cooperative Spectrum Access .....	26
Figure 2.11 Inter-networks and Intra-network in Cognitive Radio Networks .....	26
Figure 2.12 Overlay and Underlay Spectrum Sharing .....	27
Figure 2.13 Cognitive Radio Network Cross Layer Architecture .....	29
Figure 3.1 Internet of Things .....	36
Figure 3.2 Three, Four and Five Layered IoT Architecture .....	37
Figure 3.3 IoT Applications Domains .....	42
Figure 3.4 IoT transportation Application Example .....	44
Figure 3.5 Block Diagram of a Smart Healthcare System .....	45
Figure 3.6 IoT Smart Grid Application .....	46
Figure 3.7 IoT Agriculture Application .....	47
Figure 3.8 IoT Platform Main Requirements .....	48
Figure 3.9 Fog Computing Architecture .....	49
Figure 4.1 Design Factors of Cognitive Radio for Internet of Things System .....	53
Figure 4.2 Classification of Channel Assignment for CRIOts .....	56
Figure 5.1 An Example of Obtaining the Idle Set and Busy Set from the Sensed Data .....	73

Figure 5.2 Prediction System using Bayes Theorem and HMM .....	76
Figure 5.3 Support Vector Machine Hyperplane Examples .....	77
Figure 5.4 Performance Evaluation of the three Protocols Using Different Number of Hidden States .....	83
Figure 5.5 Performance Evaluation of the three Protocols Using Different Number of Training Days.....	85
Figure 5.6 Performance Evaluation of the three Protocols Using Different Matrix Window Sizes .....	87
Figure 6.1 Concept of Modification of a Searching Point by PSO .....	94
Figure 6.2 System Architecture.....	104
Figure 6.3 Block Diagram of the Proposed Protocol.....	107
Figure 6.4 Sequence Diagram of the Proposed Protocol .....	108
Figure 6.5 Performance Evaluations of the two Protocols Using Different Number of IoT devices per Fog Node.....	115
Figure 6.6 Performance Evaluations of the two Protocols Using Different Number of Channels.....	117
Figure 6.7 Performance Evaluations of the two Protocols Using Different Packets' Rate .....	119
Figure 7.1 System Architecture.....	126
Figure 7.2 Classical Round Robin Cycle .....	134
Figure 7.3 Enhanced Round Robin Cycle.....	135
Figure 7.4 Vertex Cover Cycle .....	137
Figure 7.5 Performance Evaluations of the three Access Strategies Using Different Number of RSUs per Fog Node.....	142
Figure 7.6 : Performance Evaluations of the three Access Strategies Using Different Number of Fog Nodes.....	145
Figure 7. 7 Performance Evaluations of the three Access Strategies Using Different Network Topology Connectivity .....	147

# List of Tables

Table 4.1 Comparison of the CRIOts Spectrum Sharing/Allocation Protocols .....	66
Table 5.1 Simulation Parameters' Value .....	80
Table 5.2 Percentage of Improvement using the Proposed 2-model HMM and 2-model HMM-SVM Protocols Compared to the Classical HMM Protocol .....	89
Table 6.1 Variables' notation.....	99
Table 6.2 Simulation Parameters' Value .....	113
Table 6.3 Percentage of Improvement using the Proposed SDP-PSO Protocol Compared to the Spectrum Auction Protocol .....	120
Table 7.1 Variables' Notation.....	129
Table 7.2 Simulation Parameters' Value .....	140
Table 7.3 Percentage of Improvement using the Proposed Vertex Cover and Enhanced Round Robin Access Strategies Compared to the Classical Round Robin Access Strategy .....	148

# List of Abbreviations

<b>ACSS</b>	Adaptive Cooperative Spectrum Sensing
<b>AP</b>	Access Point
<b>BS</b>	Base Station
<b>CIoT</b>	Cognitive Internet of Things
<b>CR</b>	Cognitive Radio
<b>CRIoT</b>	Cognitive Radio Internet of Things
<b>CRNs</b>	Cognitive Radio Networks
<b>CSMA/CA</b>	Carrier Sense Multiple Access with Collision Avoidance.
<b>CTS</b>	Clear-to-Send
<b>D2D</b>	Device to Device
<b>DCSS</b>	Decentralized Cooperative Spectrum Sensing
<b>DP-PSO</b>	Discrete Permutation Particle Swarm Optimization
<b>DSA</b>	Dynamic Spectrum Access
<b>ELM</b>	Extreme Learning Machine
<b>ERR</b>	Enhanced Round Robin
<b>FCC</b>	Federal Communications Commission
<b>FCFS</b>	First Come First Served
<b>GB</b>	Guard Band
<b>HE</b>	Homo Equalis
<b>HMM</b>	Hidden Markov Model
<b>IoT</b>	Internet of Things
<b>LoRa WAN</b>	Long Range Wide Area Network
<b>LPWANs</b>	Low Power Wide Area Networks
<b>LTE-M</b>	Long Term Evolution
<b>MAC</b>	Medium Access Control

<b>MLP</b>	Multilayer Perceptron
<b>NB-IoT</b>	Narrow band IoT
<b>NBS</b>	Nash Bargaining Solution
<b>NP-hard</b>	Non-Deterministic Polynomial-Time Hard
<b>PBS</b>	Primary users Base Stations
<b>PDA</b> s	Personal Digital Assistants
<b>PR</b>	Primary Receiver
<b>PSO</b>	Particle Swarm Optimization
<b>PTS</b>	Partner-to-Send
<b>PU</b> s	Primary Users
<b>QoS</b>	Quality of Service
<b>RF</b>	Radial Frequency
<b>RFID</b>	Radio Frequency Identification
<b>RR</b>	Classical Round Robin
<b>RSU</b>	Road Side Unit
<b>RTS</b>	Request-to-Send
<b>SDP-PSO</b>	Scheduling based-on Discrete Permutation Particle Swarm Optimization
<b>SIGS</b> s	Secondary Information Gathering Stations
<b>SINR</b>	Signal to Interference-Plus-Noise Ratio
<b>SLFN</b>	Single Hidden Layer Feed Forward Neural Network
<b>SU</b> s	Secondary Users
<b>SVM</b>	Support Vector Machine
<b>VC</b>	Vertex Cover
<b>xG</b>	NeXt Generation



# Abstract

Internet of Things (IoT) presents a new life style by developing smart homes, smart grids, smart city, smart transportation, etc., so IoT is developing rapidly. However recent researches focus on developing the IoT applications disregarding the IoT spectrum scarcity problem facing it. Integrating Internet of Things (IoT) technology and Cognitive Radio Networks (CRNs), forming Cognitive Radio Internet of Things (CRIoTs), is an economical solution for overcoming the IoT spectrum scarcity. The aim of this thesis is to solve the problem of spectrum sharing for CRIoT; the work in thesis is presented in three parts, each represents a contribution.

Our first contribution is to propose two new protocols to solve the problem of channel status prediction for interweave CRNs. Both protocols use Hidden Markov Model (HMM). In the training stage of both protocols, the available data are trained to produce two HMM models, an idle HMM model and a busy one. Both models are used together to produce the 2-model HMM. In the prediction stage the first protocol uses Bayes theorem and the 2-model HMM, while the second protocol uses Support Vector Machine (SVM) employing the parameters produced from applying the 2-model HMM, named 2-model HMM-SVM. The 2-model HMM-SVM outperforms the classical HMM and 2-model HMM in terms of the true percentage, the inaccuracy and the probability of primary users' collision (false negative prediction).

In our second contribution, we proposed a centralized time slotted packet scheduling protocol for CRIoTs. It uses Discrete Permutation Particle Swarm Optimization (DP-PSO) for scheduling the IoT device packets among the free slots obtained from applying cognitive radio networks' channel estimation technique proposed in the first part. Our proposed protocol is applied to smart healthcare facility. Configuring three main building blocks for the used application architecture; the IoT devices block, the first layer fog nodes block and the central fog server. Each group of IoT devices is connected to a fog node, the entire fog nodes in the system are connected to the central fog node. The proposed protocol is named Scheduling based-on Discrete Permutation Particle Swarm Optimization (SDP-PSO). An objective fitness function is formulated with three parameters; maximizing the fairness index among fog nodes, minimizing the packets' queuing delay and

minimizing the number of dropped packets that exceeded their allowed time in the network without being sent. The performance of the proposed SDP-PSO protocol overcomes an old protocol named spectrum auction in terms of the fairness between fog nodes, the average queuing delay, the number of dropped packets and the time and the space complexity.

Finally, in the third contribution, we proposed a distributed packets' scheduling protocol for CRIoTs. Our proposed protocol can be applied to an urban traffic control. The configured system in this part consists of three main building blocks; the IoT devices block, the first fog layer block (Road Side Units (RSUs)) and the second fog layer block. Each group of IoT devices is connected to a RSU, each group of RSU are connected to a fog node which acts as their cluster head. The fog nodes are connected together forming a partial mesh network. The proposed distributed packets' scheduling protocol for CRIoTs is applying three distributed access strategies together with the SDP-PSO proposed in the second part to schedule the packets on the estimated free slots resulted from applying the protocol proposed in the first part. The used access strategies are the classical round robin, in addition to two proposed ones named; the vertex cover and the enhanced round robin. An objective fitness function near similar to that used in the centralized protocol, was applied but with some differences to make it suitable for distributed scheduling. In general the vertex cover strategy performs better than both the classical round robin and the enhanced round robin strategies in terms of the fairness index among road side units, the average packet queuing delay and the percentage of dropped packets.

# Abstrait

L'Internet des objets (IoT) présente un nouveau style de vie en développant des maisons intelligentes, des réseaux intelligents, des villes intelligentes, des transports intelligents... etc., alors l'IoT se développe rapidement. Cependant, des recherches récentes se concentrent sur le développement des applications IoT sans tenir compte du problème de rareté du spectre IoT ont reconstruit. L'intégration de la technologie de l'Internet des objets (IoT) et des réseaux radio cognitifs (CRN), formant l'Internet des objets par radio cognitive (CRIoT), est une solution économique pour battre la rareté du spectre IoT. Le but de cette thèse est de résoudre le problème du partage du spectre pour CRIoT; le travail en thèse est présenté en trois parties.

Notre première contribution est de proposer deux nouveaux protocoles pour résoudre le problème de la prédiction de l'état des canaux pour les CRN entrelacés. Les deux protocoles utilisent le modèle de Markov caché (HMM). Dans la phase d'entraînement des deux protocoles, les données disponibles sont utilisées pour produire deux modèles HMM, un modèle HMM occupé et un modèle pas occupé. Les deux modèles sont utilisés ensemble pour produire le HMM à 2 modèles (2-model HMM). Dans la phase de prédiction, le premier protocole utilise le théorème de Bayes et le modèle HMM à 2 modèles, tandis que le second protocole utilise la machine à vecteur de support (SVM) qui emploie les paramètres produits à partir de l'application du modèle HMM à 2 modèles. Le HMM-SVM à 2 modèles est supérieure à l' HMM classique et le HMM à 2 modèles en termes de pourcentage réel, d'imprécision et de probabilité de collision des utilisateurs principaux (prédiction de faux négatifs).

Dans notre deuxième contribution, nous avons proposé un protocole central pour ordonner les de paquets pour CRIoT à créneaux temporels. On applique l'optimisation des essaims de particules à permutation discrète (DP-PSO) pour programmer les paquets IoT parmi les créneaux libres obtenus en appliquant la technique d'estimation de canal des réseaux radio cognitifs proposée dans la première partie. Notre protocole proposé est appliqué aux établissements de santé intelligents. On a composé trois blocs de construction principaux pour la construction d'application utilisée; Blocage des appareils IoT, blocage des nœuds de brouillard de première couche et serveur de brouillard central. Chaque groupe d'appareils IoT est

connecté à un nœud de brouillard, tous les nœuds de brouillard du système sont connectés au nœud de brouillard central. Le protocole proposé est nommé Planification basée sur l'optimisation des essais de particules à permutation discrète (SDP-PSO). Une fonction de fitness objective est formulée avec trois paramètres; maximiser l'indice d'équité entre les nœuds de brouillard, minimiser le délai de mise en file d'attente des paquets et minimiser le nombre de paquets abandonnés qui ont dépassé leur temps dans le réseau sans être envoyés. Les performances du protocole SDP-PSO proposé surmontent un ancien protocole appelé vente aux enchères de spectre en termes d'équité entre les nœuds de brouillard, le délai moyen de mise en file d'attente, le nombre de paquets abandonnés et la complexité temporelle et spatiale.

Enfin, dans la troisième contribution, nous avons proposé un protocole d'ordonnancement de paquets distribués pour CRIoT. Notre proposition de protocole est appliquée à un contrôle de la circulation urbaine. Le système configuré dans cette partie se compose de trois blocs de construction principaux: le bloc d'appareils IoT, le premier bloc de couche de brouillard (unités routières (RSU)) et le deuxième bloc de couche de brouillard. Chaque groupe d'appareils IoT est connecté à une RSU, chaque groupe de RSU est connecté à un nœud de brouillard qui est en de tête du groupe. Les nœuds de brouillard sont connectés ensemble pour former un réseau maillé partiel. Le protocole de planification des paquets distribués proposé pour CRIoT applique trois stratégies d'accès distribuées avec le protocole SDP-PSO proposé dans la deuxième partie pour planifier les paquets sur les intervalles libres estimés résultant de l'application du protocole proposé dans la première partie. Les stratégies d'accès utilisées sont le round robin classique, en plus deux stratégies proposées nommées; la couverture vertex et le round robin amélioré. Une fonction de remise en forme objective presque similaire à celle utilisée dans le protocole centralisé a été appliquée, mais avec quelques différences pour la rendre adaptée à la planification distribuée. En général, la stratégie de couverture de vertex fonctionne mieux que les stratégies à tour de rôle et à tour de rôle amélioré en termes d'indice d'équité entre les unités en bordure de route, de délai moyen de mise en file d'attente de paquets et de pourcentage de paquets abandonnés.

# **Chapter 1**

## **Introduction**

# Chapter 1

## Introduction

Internet of Things (IoT) will present a new life style by creating smart homes, smart grids, smart transportation, etc., so it will be developed rapidly. Recent IoT researches focus only on communication, computing and connectivity of IoT devices. A main challenge is neglected, which is the spectrum scarcity. The IoT bandwidth scarcity arises as a result of the enormous increase of the IoT devices numbers. Thus, it is envisioned that IoT will not grow as expected without the integration of IoT technology and the cognitive radio abilities in order to increase the spectrum bandwidth.

This chapter is an introduction to this thesis; it is divided into two sections. The first section describes the problem introduced in the thesis. The second section illustrates the thesis organization.

### 1.1 Problem Description

This section defines the thesis objectives, it consists of three subsections. The first one gives a brief explanation about the Internet of Things concept, while the second one defines the Cognitive Radio Networks and its functionalities. Finally, the last one illustrates briefly the integration between the Cognitive Radio Networks and the Internet of Things.

#### 1.1.1 Internet of Things

Internet of Things (IoT) is a world-wide network of interconnected things that allow people to be connected to things. These things can be computing devices, mechanical and machines, objects, in general any device which can be assigned an IP address, to be able to transmit data over a network. These things can be connected anytime, anyplace, with anyone and anything, using any path or any network and any service.

IoT grows rapidly, by 2025 it is expected to be consisting of about 75 billion connected things including televisions, cars, kitchen appliances, surveillance

cameras, mobile phones, utility meters, monitors, thermostats, medical devices and almost anything that we can possibly imagine.

IoT is presenting a new life style by developing various applications in several aspects in the industry, environment, healthcare, education, internet of vehicles, smart grid, smart energy management, smart security, smart farming, smart transportations, smart homes, smart building and smart cities.

### **1.1.2 Cognitive Radio Networks**

Accessing the limited unlicensed spectrum for wireless services has been increased rapidly, so the regular fixed spectrum assignment policy to the available spectrum became inconvenient. A large part of the available spectrum is allocated to licensed holders, this part is unutilized for a long time.

Cognitive Radio Networks (CRNs) is introduced, which grant its unlicensed users (named secondary users) to access and share the licensed spectrum band with its original users (named primary users). However this sharing must be without interference with their legitimate users and without any degradation in their Quality of Service (QoS).

In Cognitive Radio Network the following four functions are required:

- **Spectrum Sensing:** observing the licensed spectrum to detect its unused spectrum bands (spectrum holes).
- **Spectrum Management:** evaluating the quality of service of the detected holes and selecting the appropriate spectrum hole which satisfies the required communication requirements.
- **Spectrum Mobility:** maintaining a seamless transmission while changing the operating frequency when a primary user suddenly appears.
- **Spectrum Sharing:** preserving a fair scheduling technique for the available spectrum among the cognitive radio users.

### **1.1.3 Cognitive Radio Internet of Things**

CRNs is integrated with the IoT technology to be called Cognitive Radio Internet of Things (CRIoTs). CRIoTs will be used in many applications especially the time sensitive ones such as smart health care, smart traffic, etc. The merged

technology should satisfy many network parameters such as channel availability, reliability, energy efficiency, high throughput, etc.

Both the IoT devices and the cognitive radio's primary users are hugely increasing, so the spectrum allocation is expected to become worth, thus an efficient spectrum sharing protocol is extremely needed.

This thesis will offer a complete solution for CRIoTs scheduling protocol. It will offer three protocols, the first one will predict the empty holes in the primary network, and the second will schedule the IoT devices' packets among the predicted empty slots in a centralized way, while the third one will schedule the IoT devices' packets among the predicted empty slots but in a distributed way.

## 1.2 Thesis Organization

The thesis is composed of eight chapters

- **Chapter 1 (Introduction)**

This chapter is an introduction to this thesis; it describes the problem introduced in the thesis and states the thesis organization.

- **Chapter 2 (Cognitive Radio Networks)**

This chapter gives an overview of cognitive radio networks. It first explains the concept of cognitive radio and cognitive radio networks. Then, it illustrates various cognitive radio networks elements, together with its classification. After that, the cognitive radio networks main functions and cross layer design are illustrated. Finally, its applications, advantages and disadvantages are illustrated.

- **Chapter 3 (Internet of Things)**

This chapter gives a survey about the IoT concept in general. First, comprehensive IoT definitions were illustrated. Then, an explanation about the IoT architecture layers functions, and the most common IoT communication



technologies were presented. After that, the most famous IoT applications are overviewed. And finally, the fog computing concept -introduced to support IoT applications- is defined together with its IoT usage.

- **Chapter 4 (Spectrum Sharing/Allocation for Cognitive Radio Internet of Things)**

This chapter is considered as a survey for the spectrum sharing/Allocation in Cognitive Radio Internet of Things (CRIoTs). The chapter first defines CRIoT, together with its design factors and common challenges for CRIoTs. Then, the existing CRIoTs spectrum assignment (sharing/allocation) protocols are surveyed.

- **Chapter 5 (Cognitive Radio Networks Channel State Estimation Using Machine Learning Techniques)**

The chapter first reviews some protocols proposed for channel status prediction for CRNs. Then, it proposes two new protocols that use HMM to solve the same problem, in order to obtain the expected unused spectrum holes. Evaluations of the proposed protocols are illustrated by comparing their performance with that of the original HMM under various conditions.

- **Chapter 6 (Centralized Packets Scheduling Technique for Cognitive Radio Internet of Things Based on Discrete Permutation Particle Swarm Optimization)**

This chapter first reviews the Particle Swarm Optimization (PSO) and an old centralized scheduling method named the spectrum auction technique. Then, the problem of scheduling and resource allocation of IoT devices' packets among the free holes obtained from chapter 5; is formulated. The new centralized packets' scheduling strategy for CRIoTs is then proposed. Finally, the evaluations of the proposed protocol are presented by comparing its performance with that of the spectrum auction technique under various conditions.

- **Chapter 7 (Distributed Packets Scheduling Technique for Cognitive Radio Internet of Things Based on Discrete Permutation Particle Swarm Optimization)**

This chapter first presents a brief review about the distributed scheduling. Then, it uses three distributed access strategies -an old one and two proposed ones- along with the protocol proposed in chapter 6 to schedule the IoT devices' packets in a distributed manner. Evaluations of the proposed access strategies are presented by comparing their performance with the old one under various conditions.

- **Chapter 8 (Conclusion and Future Works)**

This chapter illustrates the thesis key conclusions, and points out some of the possible future works.

# **Chapter 2**

## **Cognitive Radio Networks**

# Chapter 2

## Cognitive Radio Networks

Cognitive Radio (CR) can be defined as an intelligent communication device which can interact with the surrounding environment in order to be aware of several things such as its radio frequency environment, its internal state, its location and the application needs, etc. Using this information the Cognitive Radio has to dynamically reconfigure itself to achieve the desirable communication objectives by adapting its operating parameters (e.g. operating frequency, transmitting power, communication technology, modulation scheme, etc.).

Cognitive Radio Networks (CRNs) can be defined as a network with dynamic licensed spectrum access capabilities. It consists of wireless users and cognitive radio. Because of the spectrum scarcity problem in the unlicensed spectrum band, the underutilization of the licensed spectrum band, and the intelligence of the Cognitive Radio; a Cognitive Radio Network uses these factors in order to acquire the ability to access and share the licensed spectrum band with its legitimate users with minimum interference with them and with no degradation in their quality of service.

This chapter overviews the cognitive radio networks; it is divided into eight sections. The first section gives an explanation about both the idea of cognitive radio, and cognitive radio networks. The second section states and gives a brief explanation about the basic elements of the cognitive radio network. The cognitive radio networks' classification is stated in the third section. The cognitive radio main functions and cross layer design are illustrated in the fourth and the fifth section respectively. The cognitive radio networks applications, advantages and disadvantages are clarified in the sixth and the seventh sections respectively. Finally, this chapter summarization is stated in the last section.

### 2.1 Cognitive Radio and Cognitive Radio Networks

Recently the usage of wireless devices has been growing in an extremely rapid rate, so accessing and using the limited spectrum band has been experiencing an enormous increase. The radio frequencies that any wireless signal uses to traverse are

split into licensed band and unlicensed band. The licensed band is a fixed portion of the frequency band that has been assigned to a wireless service or a specific holder for their usage, while unlicensed spectrum band can be used for anyone wireless transmission [1].

This traditional fixed spectrum assignment policy to wireless spectrum bands became ineffectual, as some bands experience excessive usage (the unlicensed spectrum band), while the larger portion of the spectrum are unused (the licensed spectrum band). Stating this fact, explains the increase in interference and congestion in networks that use the unlicensed spectrum band, which results in a remarkable degradation of these network throughput, while the licensed spectrum bands remain unutilized. So the commonly used wireless spectrum band access approach became inefficient [1], [2].

To solve this problem efficiently some researcher suggested allowing unlicensed wireless users networks to share the large unused portion of the licensed spectrum band networks. This type of networks first got several names such as Dynamic Spectrum Access (DSA) networks or NeXt Generation (xG) communication networks, but finally it got its lasting name; Cognitive Radio Networks. Cognitive Radio Networks (CRNs) allow the unlicensed users named Secondary Users (SUs) to access and share the licensed spectrum band in an opportunistic way under the conditions that the licensed original users named Primary Users (PUs) performance won't be affected [1], [2].

As shown in figure 2.1 [3] both primary and secondary users coexist in the same frequency band, by giving the secondary users permission to use the spectral holes existing in the licensed spectrum band, which can be defined as the frequencies which are not currently used by any primary users [1], [4], [5]. Using cognitive radio networks, the unlicensed band's interference and congestion problems are solved and the whole spectrum is utilized efficiently [6].

This section is divided into three main subsections; the first one states the history of cognitive radio, then a brief explanation about cognitive radio and cognitive radio network is explained in the second and the third subsections respectively.

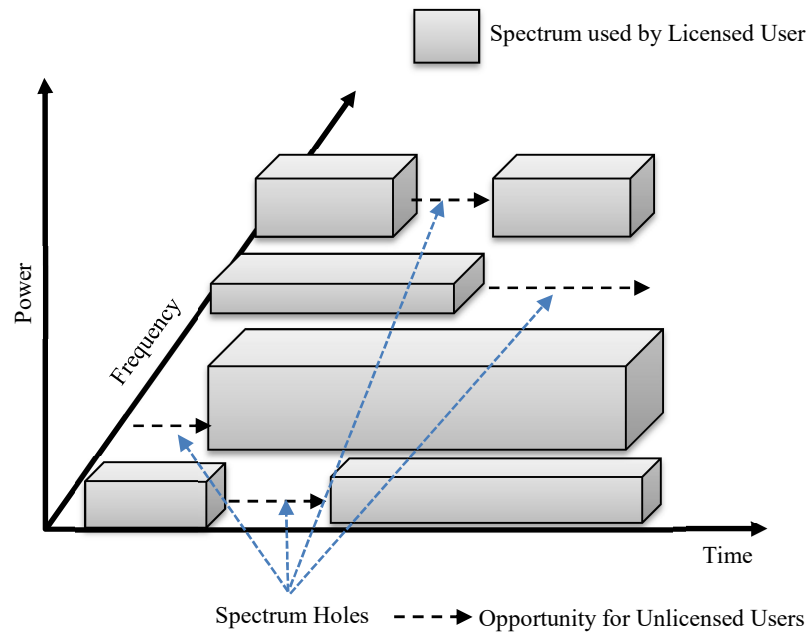


Figure 2.1 Radio Spectrum Holes Concept [7]

### 2.1.1 Cognitive Radio History

"Cognitive Radio" expression was first used by DR Joseph Mitola III while exchanging an email between himself, Jens Zander and Gerald Q. ("Chip") Maguire. In 1998 Mitola presented this term in his Licentiate lecture at KTH, the Royal Institute of Technology. Then, a year later, he illustrated the importance of the merge between the wireless technology and computational intelligence, and also interpreted the Cognitive Radio (CR) expression in his doctoral dissertation [8].

Mitola had the idea that radios, related computer-to-computer communications of wireless Personal Digital Assistants (PDAs) and the related networks should have computational intelligent capabilities, to be able to acquire wireless communication service and network resources, by self-recognizing their network needs. Although Mitola defined cognitive radio term, this term developed several other meaning over time in other context.

In the Center for Wireless Communications at the University of Oulu, on Monday the 11<sup>th</sup> of January 2010, the first cognitive radio network call was done using Cognitive Wireless Clouds' cognitive Radio Assisted Mobile Ad hoc Network that was developed by Cognitive Wireless Clouds Researchers [9].

## 2.1.2 Cognitive Radio Concept and Abilities

Cognitive Radio can be defined as software defined radio in wireless communication with some intelligent and adaptive capabilities. It has the ability to monitor, interact and communicate with its surrounding environment to be aware of several information such as radio frequency environment, internal state, location and application needs, etc. So that in real time the radio can modify its operating parameters (e.g. operating frequency, modulation scheme, transmitting power, communication technology, etc.) and performing self-adjustment to fulfill its communication target [1], [10]–[12].

Several papers address the functionalities that allow cognitive radio to work efficiently. Collecting the functions mentioned by researchers in [10], [11]; cognitive radio should be able to explore its surrounding operating environment to gather communication information about the surrounding operating radio frequencies, location and application needs. Also, it should be aware of the external situation and internal state. Cognitive radio should communicate and interact with the surrounding local radio and emissions operations by peregrination across boundaries and reconfigure itself. Furthermore cognitive radio must be able to detect the unused portion of the licensed spectrum band by continuously performing local sensing of the spectrum. Cognitive radio must be intelligent enough to evaluate the communication status in contrast with its communication targets and capability. It should also be able to negotiate with the service provider to obtain the best available channel with the lowest price for its secondary users. Also it must perform self-adjustment to lower its power consumption. Cognitive radio must be aware of the networks users' location to reduce primary and secondary users' interference seamlessly. Likewise cognitive radio must record their users' choices and actions to be able to estimate their needs. And finally cognitive radios must cooperate together by integrating gathered information and executing commends transmitted by others.

Due to the cognitive radio functionalities mentioned above; the main challenge is its coexistence with the licensed network as well as the guarantee of the licensed network's quality of service [4].

### **2.1.3 Cognitive Radio Network Definition**

Cognitive Radio Network can be defined as a network that consists of smart cognitive radios wireless users, and dynamic spectrum access capabilities [2]. The cognitive radio networks was first introduced while trying to solve the problem of spectrum scarcity and network congestion faced when using the unlicensed spectrum band. Using the smart cognitive radio communication devices and the fact of spectrum scarcity in unlicensed spectrum band, a cognitive radio network was introduced, in order to give the cognitive radio users; named secondary users (SUs); the permission to access and share the licensed spectrum band with its original users; named primary users, but with the guarantee that the original users performance won't be affected [1], [6].

It is clear that a cognitive radio network doesn't have a specific spectrum band assigned to it, but it coexist with one or more primary networks in their licensed spectrum band [1]. It can access and share the licensed frequency channels without or with minimal interference with the primary networks' users so the primary users quality of service won't be degraded. Cognitive radio networks may detect the unused portion (channels) of the licensed spectrum band by detecting the primary users' transmission, and then assign the unused channels to its users [13].

Some cognitive radio features are added to the network and above layers functionalities in order to form a cognitive radio network [10].

## **2.2 Cognitive Radio Networks Elements**

Cognitive radio networks concept is practically a promising communication technology, proposed to overcome the spectrum scarcity and interference problem faced when using the unlicensed spectrum band. Employing such technology requires setting several heterogenic spectrum policies and communication architecture for the already existing networks. Figure 2.2 describes the architecture of the cognitive radio networks and its components that also describe all the possible existing scenarios [1], [4], [14].



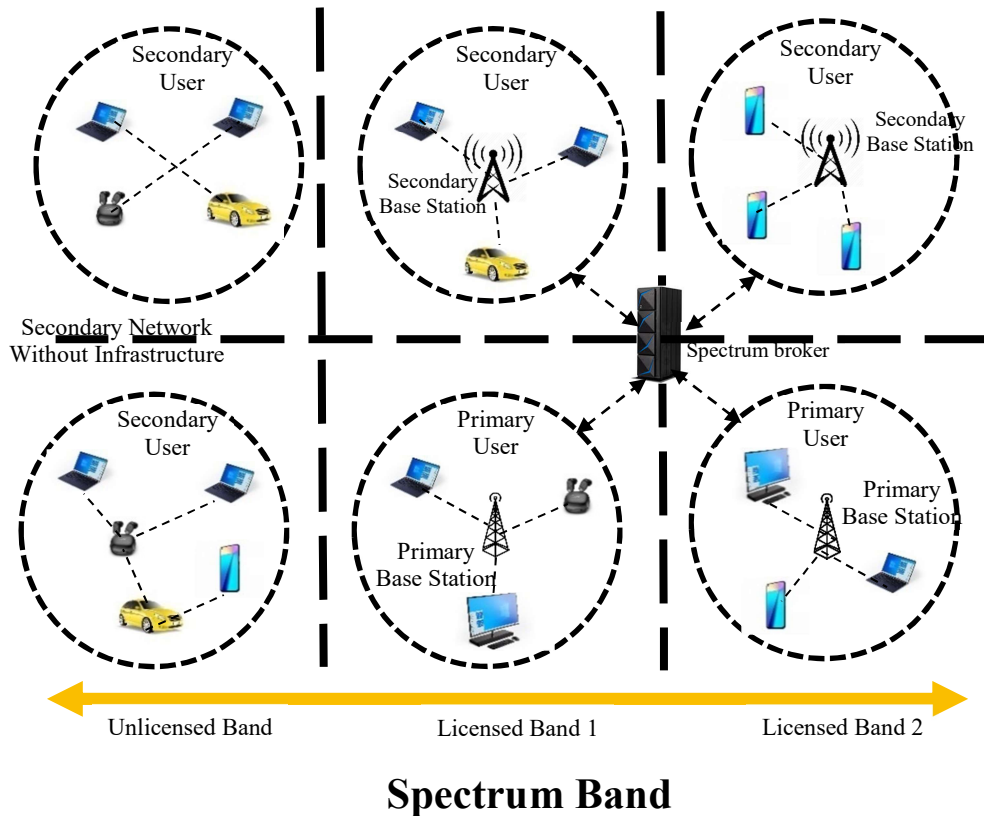


Figure 2.2 Cognitive Radio Network Architecture [14]

As shown in figure 2.2, that represents the cognitive radio networks architecture, it is divided into two main parts; one for the primary network and the other for the secondary network.

### 2.2.1 Primary Network

This is the network with licensed spectrum band, which is a fixed portion of the frequency band that has been assigned to a wireless service or a specific holder for its usage. This network has an already constructed infrastructure to serve its wireless users. The most famous examples for this network are the TV broadcast networks and the cellular networks. As shown in figure 2.2 the primary network consists of its legitimate users that are called primary users and the primary base station that serves the primary users [1], [4], [14]–[16].

- **Primary Users;** they are the legitimate or original users of the licensed network, they can also be called the licensed users. They have the right to operate and access the portion of the frequency band assigned to the primary network;

according to the permission given by the primary network base station, which control all the users' communications procedures.

- **Primary Base-Station;** it is the licensed spectrum band base station. Generally it is concerned with only the primary users' communication operations. But in few licensed base stations, it is required to install some cognitive radio protocols upon request for cognitive networks' access explained in the next lines.

## 2.2.2 Cognitive Radio Network

Cognitive radio networks are those networks that don't have a specific frequency band specified assigned to it, instead they operate opportunistically on the licensed networks' band. As shown in figure 2.2, they can operate on both infrastructure networks and ad hoc networks. The components of the Infrastructure cognitive radio network are cognitive radio users, cognitive radio base station and spectrum broker [1], [4], [14]–[16].

- **Cognitive Radio Users;** not all the researchers name them cognitive radio users or cognitive users, some refer to them as Secondary Users (SUs), others refer to them as unlicensed users. They don't have a specific spectrum to operate on it so they must have additional abilities to be able to coexist with the primary users without interfering with their signals.
- **Cognitive Radio Base-Station;** some researchers referred to it as unlicensed base station. It is a fixed infrastructure cognitive radio network element but with additional capabilities to be able to perform the cognitive radio networks functionalities. It also can act as a hub that gathers information from secondary users. It communicates with the secondary users to connect it with other networks.
- **Spectrum Broker;** it is the scheduling server that acts as spectrum information manager. It is responsible for connecting different cognitive radio networks, to share and distribute the spectrum resources among them to be able to coexist together without interfering with each other's.

## 2.3 Cognitive Radio Network Classification

Cognitive Radio Networks are classified based on architecture. The most popular network classification is based on how the network operation is done, it is either centralized or distributed or hybrid.

### 2.3.1 Centralized (Infrastructure) Architecture

In this type, there exist an infrastructure backbone network (figure 2.3), the covered area is divided into cells; each cell has a Base Station (BS) or an Access Point (AP) that acts as the central entity of the cell. It manages the secondary users' communication operations. All the secondary users' communications are done through the central entity even if both secondary users are within each other's communication range. For communication through different cells, routing is done through the backbone network. The central entity must contain several communication protocols to be able to achieve the various requirements of the secondary users [12], [15], [17]. Note that the secondary users share information and can access the cognitive radio base station on both licensed and unlicensed spectrum band.

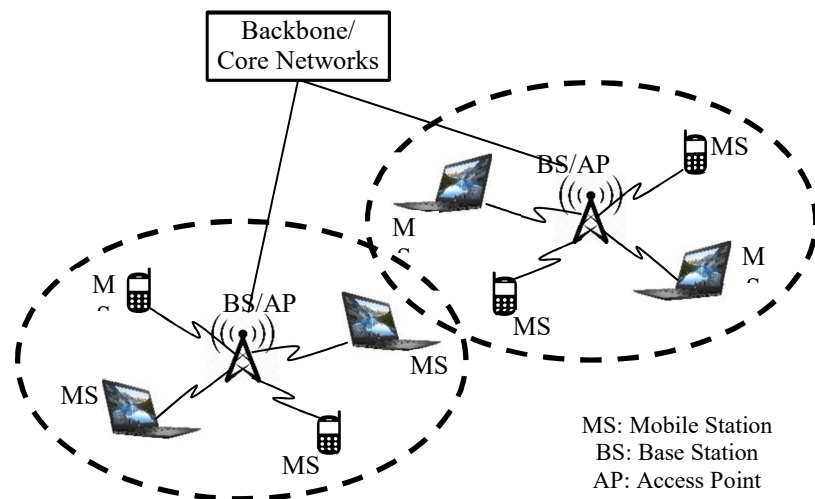


Figure 2.3 Infrastructure Architecture [12]

### 2.3.2 Distributed (Ad hoc) Architecture

In ad hoc cognitive radio network there is no infrastructure, instead an ad-hoc communication network is set up between various secondary users, by recognizing each other's existing in the same communication range as shown in figure 2.4. Any secondary user shares its information with the users in its communication range through establishing links, setting up an ad-hoc network range. Then, the information is transmitted to other ad hoc networks through multi hop communication techniques. Naturally in ad hoc networks different communication technologies may be used to form links between its users. All the cognitive radio capabilities must be installed in each secondary user, to gather information and be aware of its surrounding environment, to determine the action needed to be taken [12], [15], [17]. Note that an ad hoc communication network is used by the secondary users to share information and access both licensed and unlicensed spectrum band.

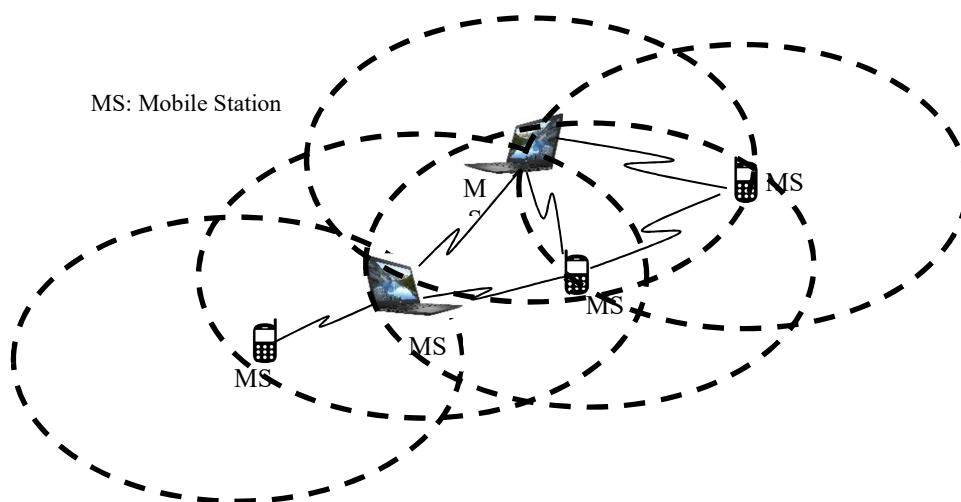


Figure 2.4 Ad-Hoc Architecture [12]

### 2.3.3 Hybrid (Mesh) Architecture

The hybrid or mesh cognitive radio networks architecture has a hybrid architecture between infrastructure and ad hoc architectures, or between centralized and distributed architecture, which is much similar to ordinary hybrid wireless mesh networks. In this architecture the needed cognitive radio network functionalities are embedded in the central entities. The central entities which are either base stations or access points, act as wireless routers, they can communicate directly between each

other. Although they form a wireless backbone, some of them may communicate with wired links and act as gateways. As shown in figure 2.5; the secondary users communicate with the central entity either directly through one hop communication or through multi hop communication using other secondary users. This architecture is more flexible and may cost less if the locations of the central entities are well planned as they may not be connected to any wired backbone network [10], [15].

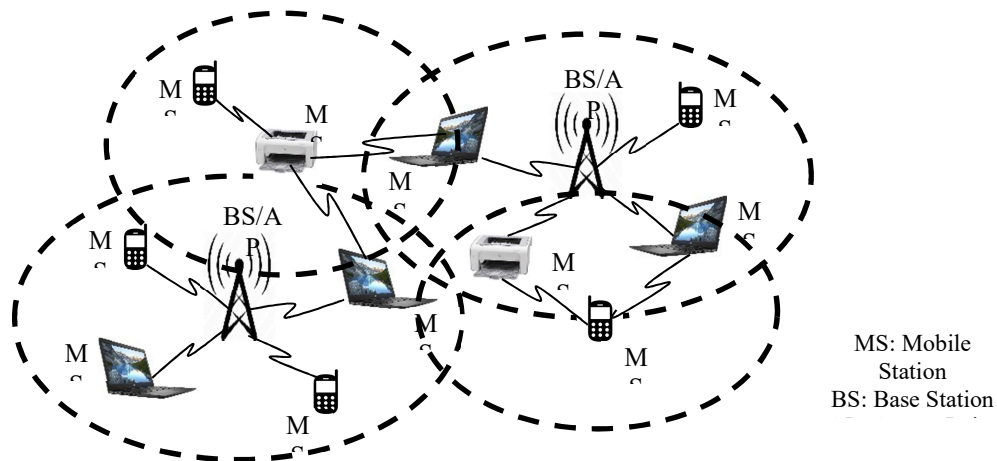


Figure 2.5 Mesh Architecture [12]

Note that in all the above types, an advanced Medium Access Control (MAC) protocol is needed for the secondary users (SUs) to access many primary networks with different MAC protocols.

## 2.4 Cognitive Radio Main Functions

Cognitive radio networks must perform the following functions:

- Access and share the licensed spectrum band with the primary networks.
- Be aware of its surrounding environment to detect the spectrum holes.
- Stay alert to detect the appearance of any primary user in order to evacuate the channel immediately and change the operating of the secondary user using the channel.

All these functions are not ordinary network functions, thus extra functionalities must be installed for the cognitive radio networks, to adapt itself to dynamic radio environment by making suitable changes in the communication

parameters [1], [4], [23]–[25], [5], [16]–[22]. Cognitive Radio functions are mostly divided into four main functions, Spectrum Sensing, Spectrum Management, Spectrum Mobility and Spectrum Sharing.

Figure 2.6 represents the cognitive radio cycle, including the four cognitive radio networks function [22], [25], [26], [27]. Each of these functions is a big area for research, in the following four subsections these functionalities will be explained and classified.

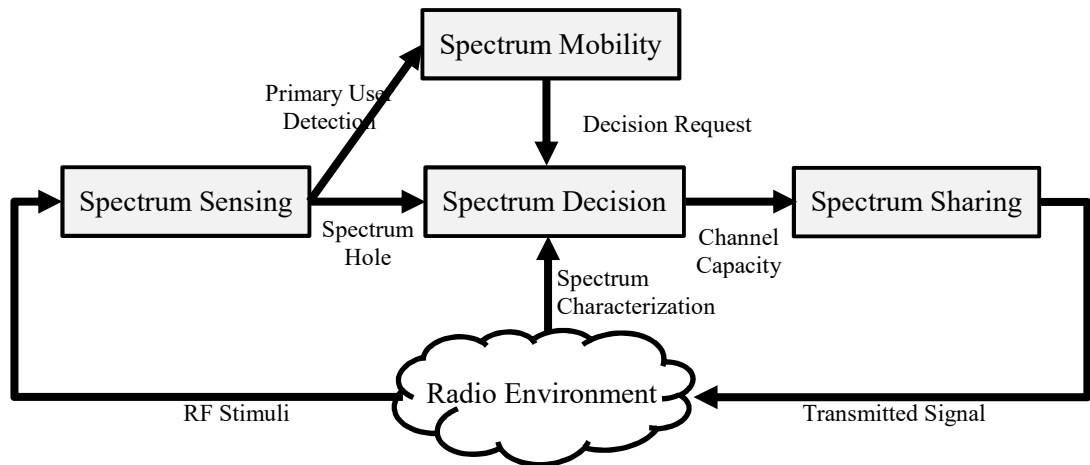


Figure 2.6 Cognitive Radio Cycle [27]

### 2.4.1 Spectrum Sensing

Spectrum sensing can be defined as the most important function performed by the cognitive radio. For the cognitive radio to be aware and detect any change in its surrounding environment, spectrum sensing must be done. Spectrum sensing is responsible for monitoring the surrounding licensed spectrum band, gather its information, then based on this information it determines the unused portion of the spectrum name holes as shown in figure 2.6 [22], [25], [26]. But detecting the spectrum holes is not a simple task, as sometimes it is not easy to detect the primary users' transmission. This task must be done without interfering with the primary users' transmission [1], [4], [23]–[25], [5], [16]–[22].

Several researches classified the spectrum sensing techniques into three categories; transmitter detection, cooperative detection and interference-based detection.

## A. Transmitter Detection

Transmitter detection is also named non-cooperative detection. In this type of detection the secondary users have the ability to determine the existence of any primary user's signal. The secondary users perform local sensing independently to detect the presence of any primary users signal. Transmitter detection have three main schemes as follows [1], [4], [19]–[23]:

- ***Matched Filter Detection***; which is classified as the optimal method for primary users signal detection if the secondary users already have the primary users' transmitted signal features. Optimal detector is used in stationary Gaussian noise. This is done by maximizing the received signal signal-to-noise ratio, then matching both the matched filter and the reference signal. The primary user received signal is demodulated, and then handed as an input to the matched filter detector to extract the information needed to make the matching process. The prime advantage of the matched filter detection is that it takes less time to acquire high processing gain due to typical coherent detection. If the known primary user's signal features are not precise, this will lead to poor matched filter detection.
- ***Energy Detection***; also known as non-coherent detection. This detection scheme is used when the secondary users can't acquire enough information about the primary users' signal features. It operates by measuring the strength of the received signal energy to determine whether the sensed channel is vacant or in use. This done by filtering the signal by a band pass filter, then the output is squared and integrated over observations of a specific time interval. And finally the integration function output is threshold with a specific predetermined threshold to determine the existence or the absence of the primary users' signal. Although this method is simple, but its main disadvantage is the inability to distinguish between the primary users' signal and unintended signals (even noise).
- ***Cyclostationary Feature Detection***; which needs prior partial information on the PUs. It makes use of the periodicity in auto-correlation of modulated the signals, it uses cyclic correlation function. Since the noise is not correlated, it uses statistical approach to differentiate noise from signals, which is its main advantage. But its main disadvantage is the complex computations and longtime observations needed to perform properly.

In most of the cognitive radio networks architecture the primary network and the secondary one are separated. Also sometimes it is impossible for the secondary users to allocate the primary user, as most of the times there is no signaling between primary and secondary user as shown in figure 2.7(a) [1], [4], [5], [20]. Because of channel fading and shadowing shown in Figure 2.7(b), secondary users may not be able to detect the primary users' signals. So it can't avoid the hidden terminal problem which is a major problem in the transmitter detection techniques. Because of the above, the listed transmitter detection techniques, so other detection schemes were introduced to solve these problems.

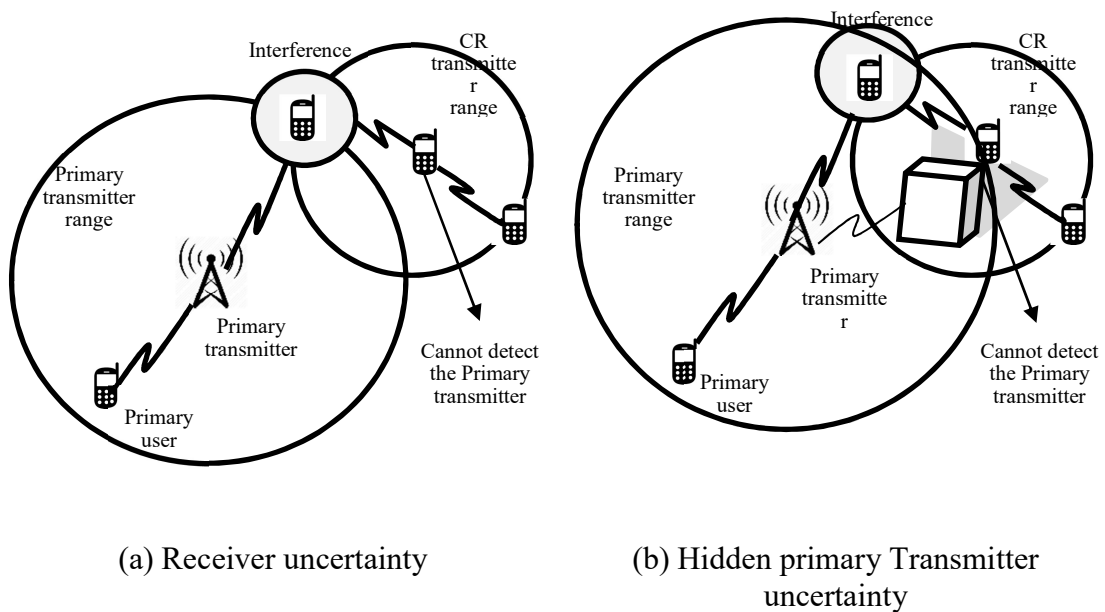


Figure 2.7 Transmitter Detection Problems [4]

## B. Cooperative Detection

This type of detection tries to overcome the problem of fault detection that arises in case of sensing information done by each secondary user independently. Instead, it gathers this information from each SU, then merge and cohere it to make the correct decision about the absence or the existence of primary users' signals. But the major disadvantage of the cooperative detection techniques is the extra network load used for the secondary user to cooperate together to perform this detection scheme, which consequently affects the network resources. Cooperative detection techniques can be classified whether it operate in a centralized or distributed manner [1], [4], [5], [19]–[25].



- **Centralized Cooperative Detection;** these techniques must have a central entity (base station/access point) that communicates with all the secondary users to gather the sensed information from them. This central entity combines all this information to make the decision whether a primary user signal exist or not.
- **Distributed Cooperative Detection;** these techniques; also named decentralized cooperative detection; don't have any central entity to gather the sensed information; however the network secondary users exchange their observation between each other.

### C. Interference-based Detection

This model was proposed by the United States Federal Communications Commission (FCC). It is shown in figure 2.8 [1], [5], [20], [28]. Its main aims are to measure and regulate the interference. The model ensures that the primary user to the secondary user interference does not exceed an interference temperature limit. It is intended to be used so that the received power level is close to that of the noise floor. Note that the noise floor level varies according to the service area. The main problem of this model is to accurately defining the interference temperature limit [1], [4], [5], [20].

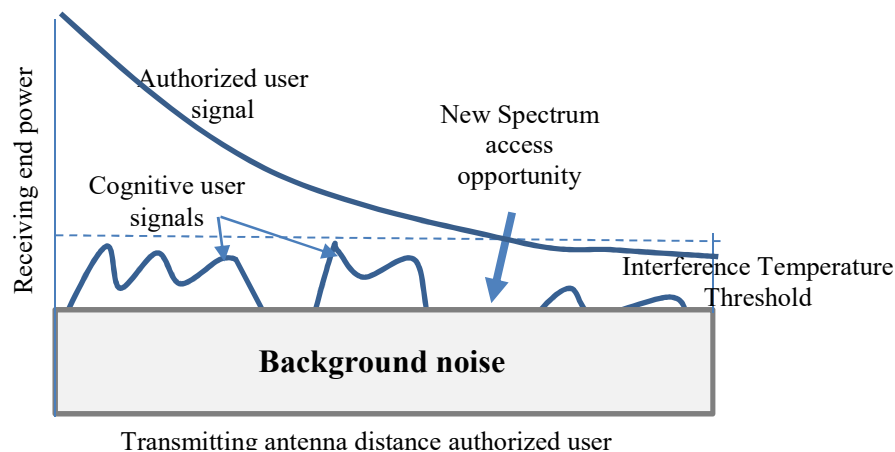


Figure 2.8 Interference Temperature Model Spectrum Management [28]

## 2.4.2 Spectrum Management

Spectrum management function gives the chance to the secondary users to communicate on the best available channel. Its main function is first to assess the

detected spectrum holes passed from the spectrum sensing function to estimate their quality, as shown in figure 2.6 [22], [25], [26]. Then, it compares the secondary users' requirements with the entire unused spectrum band of the licensed and unlicensed spectrum band to choose the best ones. Finally, the best match is done between the available holes properties and the secondary users' required quality of services. Spectrum management is divided into the following [1], [4], [23]–[25], [5], [16]–[22]:

#### **A. Spectrum Analysis**

Cognitive radio networks spectrum analysis function is responsible for estimating the parameters of the spectrum holes detected through the spectrum sensing function. This must be done regularly as each spectrum hole's characteristics vary change overtime.

#### **B. Spectrum Decision**

Cognitive radio networks spectrum decision function is responsible to fulfill the secondary users' communication requirements, which is not an easy task as the dynamic access of the frequency spectrum band increases, this may lead to congestion. By the knowledge of both the secondary users' communication quality of service requirements and the available channels parameters, spectrum decision selects the suitable spectrum band for each secondary user.

### **2.4.3 Spectrum Mobility**

As defined above, cognitive radio networks allow the secondary users to access and share the licensed spectrum band, but without interfering with its original primary users, and if any primary user appears the secondary user must evacuate the channel immediately. The function of dynamically changing the secondary user current operating frequency to the best available channel is called spectrum mobility. This change happens whenever a primary user appears or if the operating channel's parameters degrade. Spectrum mobility is like the handoff in ordinary networks but with different conditions as shown in figure 2.6 [1], [4], [23]–[25], [5], [16]–[22].

- ***Spectrum handoff***; Cognitive radio networks spectrum mobility is another type of spectrum handoff. As the new channel on which the secondary user jumps has

different parameters than the old one, so the transmitting parameters must also be changed to fit the new operating channel. The secondary user must adapt itself to the new operating frequency quick and seamless, so its transmission won't be affected. The management of spectrum mobility is responsible for managing seamless communication requirements and smooth transition during changing operating channel, so the application currently running at the secondary user won't be affected during the handoff and so experience minimum degradation in the performance. The spectrum sensing techniques must inform the spectrum mobility management with the handoff duration, so the spectrum mobility management maintains seamless and smooth transition with minimum degradation in the secondary users' currently transmitting applications [5], [12], [15], [23].

#### **2.4.4 Spectrum Sharing**

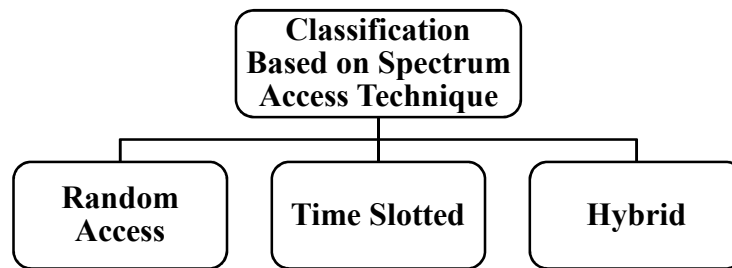
As there exist a huge number of secondary users which all have the right to access the detected frequency band, there must be a function that coordinate their spectrum access. Spectrum sharing provides a scheduling technique to divide the available spectrum band among the current secondary users as shown in figure 2.6. This function is corresponding to Medium Access Control (MAC) protocol in regular networks, but with some major different challenges as Cognitive radio network is dealing with different network conditions. The two major different challenges are the dynamic nature of the current available spectrum band and the existence of two different sets of users (primary users and secondary users).

Several research papers introduced different classifications for spectrum sharing techniques; based on technique, based on architecture, based on spectrum allocation behavior, based on scope and based on spectrum access time. Figure 2.9 illustrates these classifications. In the following subsections; these techniques various classifications will be explained.

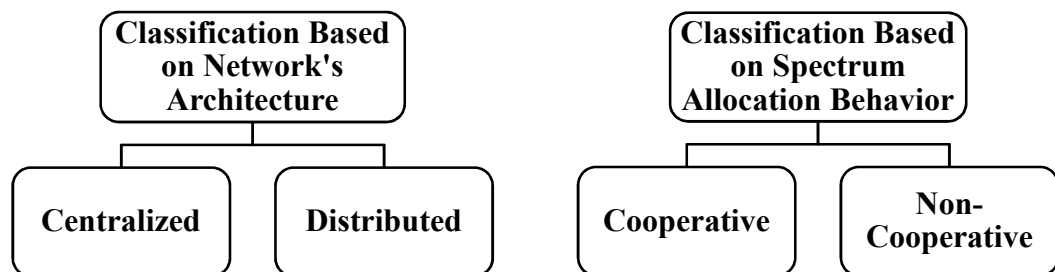
##### **A. Classification Based on Spectrum Access Technique**

Cognitive radio networks Medium Access Control protocols can be classified based on its used access method; random access MAC protocols, time slotted MAC protocols and hybrid MAC protocols (figure 2.9 (a)) [29], [30].

- **Random access MAC protocols;** the users in these protocols don't have to be synchronized with the network. Most of these MAC protocols are developed based on Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA).
- **Time slotted MAC protocols;** the users in these MAC protocols must be synchronized with the network. Time is split into slots, which are divided between the users. Some slots are used for control and others for data.
- **Hybrid MAC protocols;** this type of MAC protocol is a combination between the random access and the time slotted protocols. For example; the control signals transmission is done using the time slotted MAC scheme while the data transmission signals is done using the random access MAC scheme.

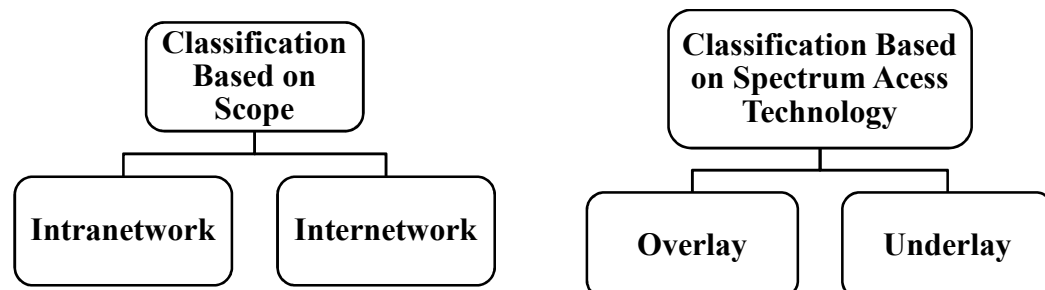


(a) Based on Spectrum Access Technique



(b) Based on Network Architecture

(c) Based on Spectrum Allocation Behavior



(d) Based on Scope

(e) Based on Spectrum Access Technology

Figure 2.9 Classifications of Medium Access Control Protocols in Cognitive Radio Networks

## B. Classification Based on Network's Architecture

Cognitive radio networks Medium Access Control protocols can be classified based on the network's architecture or infrastructure as either centralized MAC protocols or distributed MAC protocols (figure 2.9 (b)) [1], [4], [30], [31].

- **Centralized MAC Protocols**; these types of MAC protocols must have a central entity; a base station or an access point; to perform the spectrum access/ allocation process. The central entity collects all the secondary users' allocation communication requirements, then uses the available channels parameters to construct the allocation map.
- **Distributed MAC Protocols**; these types of MAC protocols don't have any central entity, instead each secondary user performs the spectrum access/allocation process independently based on the local or probable global policies. These MAC protocols are used in case of infrastructure less networks, e.g. cognitive radio ad-hoc networks.

## C. Classification Based on Spectrum Allocation Behavior

Cognitive radio networks Medium Access Control protocols can be classified according to spectrum allocation behavior; cooperative MAC protocols and non-cooperative MAC protocols (figure 2.9 (c)) [1], [4], [30], [32], [33]. Figure 2.10 shows an example of cooperative and non-cooperative environment [33].

- **Cooperative MAC Protocols**; also be called collaborative MAC protocols, where each node shares its interference measurements among other nodes. There are centralized and decentralized cooperative MAC protocols. Cooperative MAC protocols may have to construct network clusters to share the nodes' interference information locally, where the local processes are done in either a centralized or a distributed way.
- **Non-Cooperative MAC Protocols**; also called non-collaborative or selfish MAC protocols. A non-cooperative MAC protocol doesn't take into consideration other nodes interference measurements and needs, instead it only considers the node at hand.

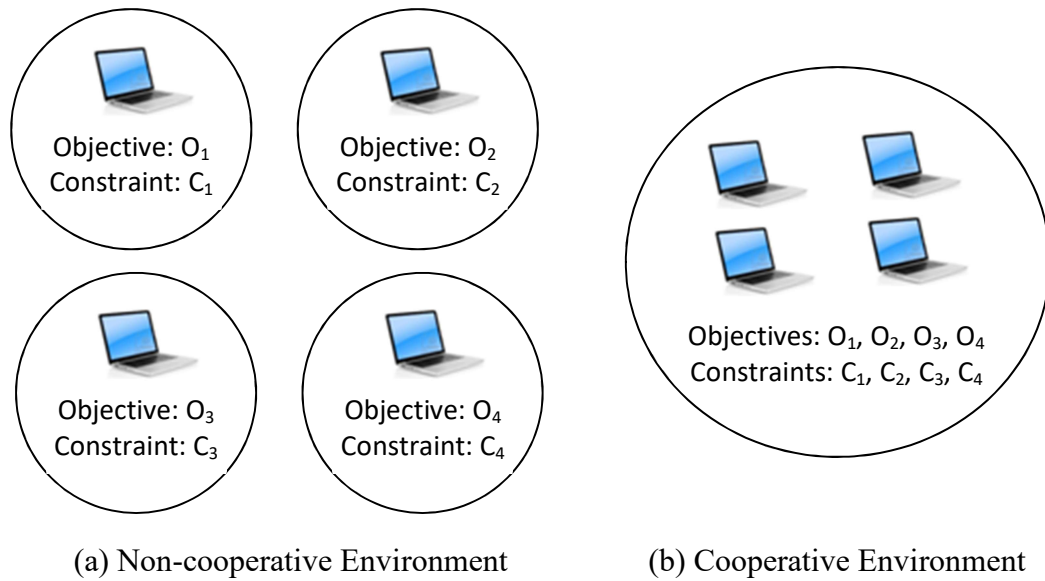


Figure 2.10 Cooperative and Non-Cooperative Spectrum Access [33]

#### D. Classification Based on Scope

Cognitive radio networks Medium Access Control protocols can be classified based on scope as intranetwork MAC protocols and internetwork MAC protocols (figure 2.6 (d)) [4].

- **Intranetwork MAC Protocols;** these types of MAC protocols distributes the available spectrum among secondary users in a single cognitive radio network (figure 2.11).

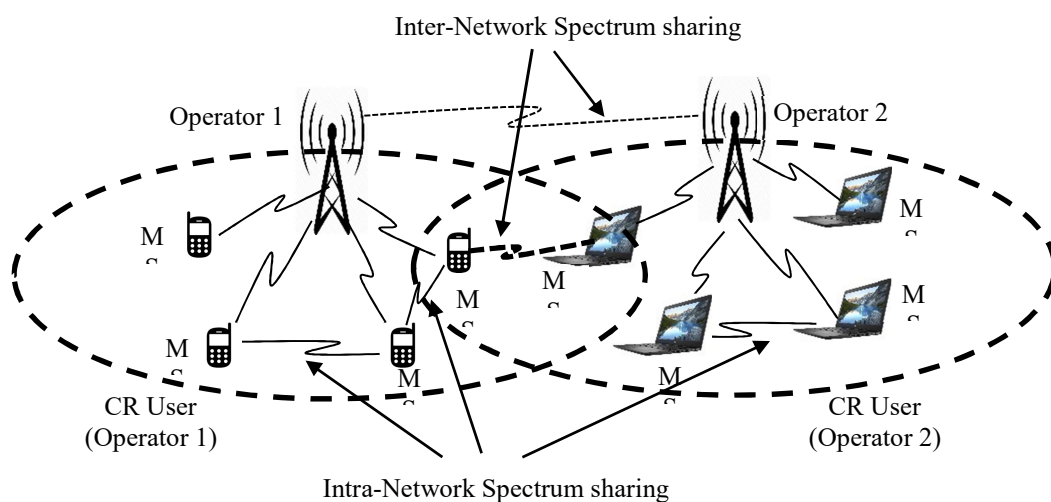


Figure 2.11 Inter-networks and Intra-network in Cognitive Radio Networks [4]

- **Inter-network MAC Protocols;** these types of MAC protocols grant different cognitive radio network to execute spectrum allocation and overlapping (figure 2.11). They include different policies about different operators, to be able to perform spectrum overlapping among different cognitive radio networks.

### E. Classification Based on Spectrum Access Technology

Cognitive radio networks Medium Access Control protocols can be classified based on spectrum access technology as either underlay or overlay MAC protocols (figure 2.6 (e)) [1], [4], [30], [32], [34].

- **Overlay MAC Protocols;** in these type of MAC protocols the secondary users use the empty parts unused by the primary spectrum band as shown in figure 2.12 (a). Of course with no or minimum interference with primary band users, meaning that any secondary user should evacuate the channel immediately in case of appearance of a primary user.
- **Underlay MAC Protocols;** in these types of MAC protocols the secondary users transmit simultaneously with the primary spectrum band, according to a spectrum allocation map, but with low transmission power that can be considered as a noise to the primary users (figure 2.12 (b)). Although the underlay MAC protocols outperform the overlay MAC protocols as it increases bandwidth utilization, but it needs complex spread spectrum techniques.

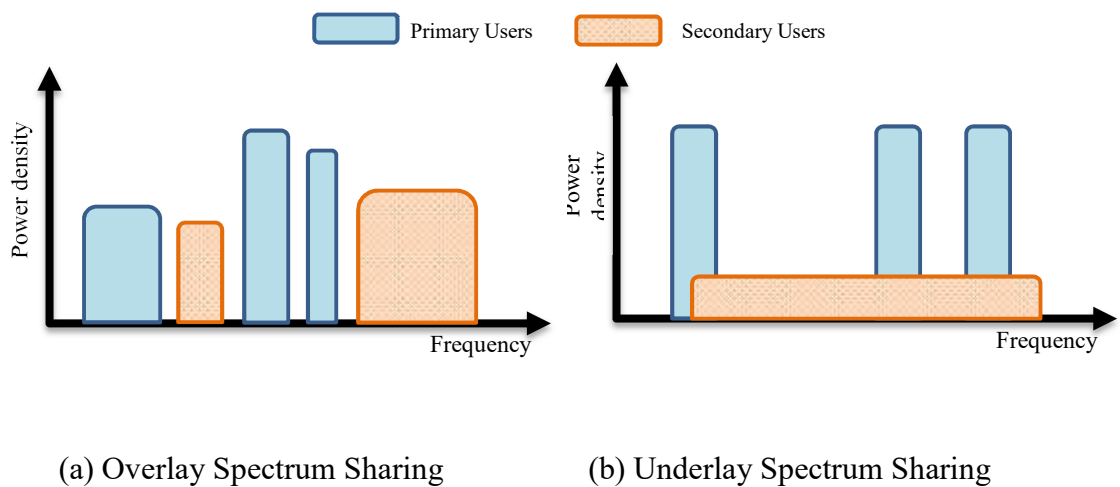


Figure 2.12 Overlay and Underlay Spectrum Sharing [34]

## **2.5 Cognitive Radio Layer Design**

The layered protocol stack is known for ordinary wired communication system, where each layer has a protocol or set of protocols developed to perform the layer functions together with only its adjacent layers. However in a wireless communication system strictly layered protocol can't be used as in wired communication system because of several reasons such as the simultaneous medium access done by several users, fading which result in small-scale variation in channels, variation in the feature of the medium over time. Also impairments intended for wireless communication that arise from signal propagation in several path; such as multipath, reflection, Diffraction, scattering, Doppler shift, etc. All these factors increase the channel features' variations [1], [4], [18], [25], [35], [36].

Cross layer design is used in wireless communication systems because of the above reasons, where it allows communication and parameters exchange with non-adjacent layers. Using this design we can view the radio system and its environment.

No definite design architecture exists for the cross layer design, so any of the cross layer designs may be used for developing cognitive radio layer design [29]. As shown in figure 2.13, the cognitive radio network functions in cross layer design need several layers cooperation to perform the required functions [1], [4], [18], [25], [35]–[37]. Each of the cognitive radio networks functions (explained in the previous section) together with its involvement in the cognitive radio cross layer design will be illustrated in the next subsections.

### **2.5.1 Spectrum Sensing Functions' Involved Layer**

Normally the physical layer performs this task, however extra instructions is required from the link (MAC) layer for the physical layer to select suitable spectrum hole from the several holes detected.

### **2.5.2 Spectrum Sharing Functions' Involved Layer**

The link layer (MAC) is the one responsible for this function, but it needs some information from the physical layer, to provide it with the channels' status and any primary user appearance.



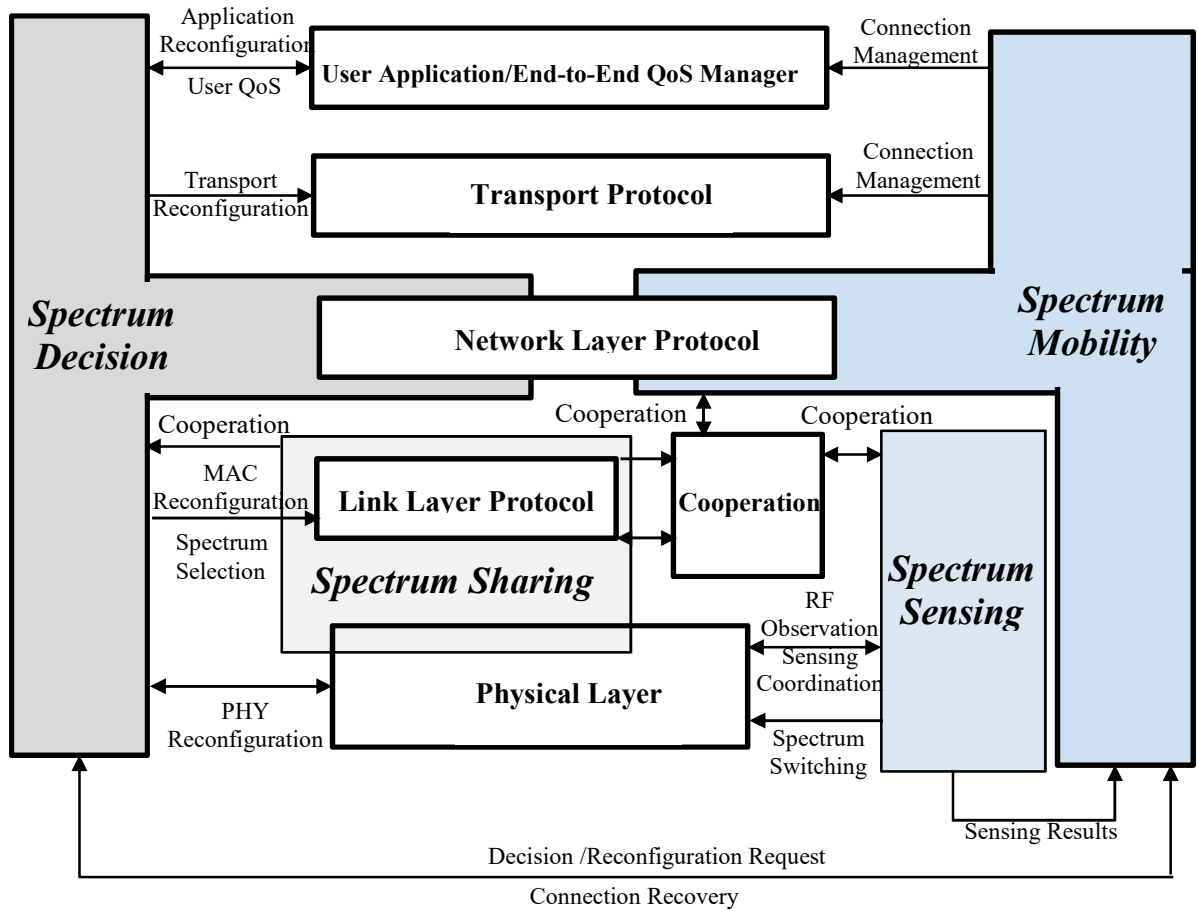


Figure 2.13 Cognitive Radio Network Cross Layer Architecture [37]

### 2.5.3 Spectrum Management Functions' Involved Layers

No specific layer is mainly responsible for performing this function, instead it requires full cooperation between all stack layers. Sensed information is sent from the physical layer to match the channel parameters with the secondary users' application's requirements. Then, the current running application analyses the sent sensed information to determine the spectrum holes that will be used. Also the MAC layer helps performing this by providing the scheduling information. After that the same reconfigurations in the transmission parameters is done by the transport layer. Then, both the link and the physical layer are reconfigured to handle the reconfigured transmission parameters.

## 2.5.4 Spectrum Mobility Functions' Involved Layers

This function also has no specific layer mainly responsible to perform it, like the spectrum management function it needs full layers cooperation. The physical layer senses the medium for the primary user appearance, then the spectrum management function takes the decision for the next spectrum holes that will be used. Then, the MAC layer calculates the assumed transition delay. After that the network layer determines the new route that will be used for the rest of the transmission. And finally the application layer attempts to cope with the lost data or the sudden degradation in the quality of service caused due to this frequency transition.

## 2.6 Cognitive Radio Networks Applications

Several applications use cognitive radio networks. This section gives a brief explanation of each of these applications:

- ***Military Networks***; this application is the most important one. Military transmits sensitive data, it is difficult to secure the military networks from any jamming or hacking. So Military uses cognitive radio networks because of its flexible approach of operating on different frequency bands. Note that, the US department of defense has developed systems to use cognitive radio techniques such as SPEAKEAS radio system [1], [17], [38], [39].
- ***Disaster Relief and Emergency Networks***; with the appearance of any disaster such as earthquakes, hurricanes, wild fires, or any great accidents like mining accidents, etc., partial or complete networks disable may happen. The main problem in such emergency situations is to allocate a network(s) to send and receive vital and critical information, so cognitive radio networks can be used to set up such emergency networks. Thus it allows emergency network to access the licensed spectrum band in an opportunistic manner, to have a significant amount of spectrum bandwidth to transmit critical reliable information [1], [17], [39].
- ***Public safety networks***; police officers, firemen and paramedic use public safety networks to communicate. Given the importance of these transmissions, the used networks can't suffer from lack of spectrum. Thus public safety network can use

cognitive radio networks approach to opportunistically access a significant amount of allocated licensed spectrum band [1], [39], [40].

- ***TV White Spaces;*** TV white space is the spectrum in the VHF/UHF bands, which is already preserved to primary users systems e.g. TV broad casters, however TV white space are not heavily used as before, so cognitive radio networks can benefit from it [41]. A great research was done in this area, and TV white space has been used as a licensed band in the cognitive radio networks by several major institutes. After the huge research done in this area; the rules of using the TV white spaces were released by the Federal Communications Commission [38].
- ***Cognitive Mesh Networks;*** because of wireless mesh networks cost effectiveness for the broadband connectivity and for the last-mile Internet access, it has been very popular approach. However the wireless band shortage has been a huge challenge. Thus if the mesh networks used the cognitive radio networks technology, it will be able to access a large portion of the wireless spectrum band opportunistically. Cognitive mesh networks can used urban areas with resources shortage [1], [39].
- ***Leased Networks;*** the primary networks can benefit from their assigned licensed spectrum portion, by making an agreement to lease a portion of it to the cognitive radio networks allowing the secondary user to access this portion of the spectrum opportunistically. However this must be done without any scarification to the primary users' quality of services. Such leasing can be applied by renting a portion of the primary network spectrum band to a mobile virtual networks operator or to a regional community for broadband access purpose [1], [39].
- ***Cellular Networks;*** recently cognitive radio networks approach has been used in cellular networks. Some efforts were made to solve the indoor coverage problem, and also to adapt to the problem of the continuously growing of communication traffic. With the usage of femtocells with cognitive radio networks approach, a femtocell searches the spectrum band to estimate the spectrum holes, so that it preserves coverage and avertes interference with any other cell either femtocells or macrocells [38].

## **2.7 Cognitive Radio Networks Advantages and Disadvantages**

From the sections explained above it can be understood that the cognitive radio networks have several properties that leads to several advantages and disadvantages. They will be listed in this section [1], [42].

### **2.7.1 Cognitive Radio Networks' Advantages**

Using cognitive radio networks have many advantages, here are the main ones:

- It can dynamically reconfigure itself according to the current transmission conditions.
- Minimize the unused band portion, so increases the spectrum band efficiency.
- The scheduling process it performs allows its secondary users to access the licensed spectrum band, so reduces their probability of collision and helps them to easily achieve their application required quality of service.
- Allow the primary network to invest in the licensed portion of the spectrum it has paid for it by leasing a portion of it to the secondary users.
- Decrease the amount of money the secondary users should pay to obtain good communication with acceptable quality of service.

### **2.7.2 Cognitive Radio Disadvantages**

As mentioned in the previous subsection, the cognitive radio networks have several advantages, but also it has several disadvantages. Here are the main ones:

- The licensed spectrum band must be used without primary users' degradation of their communication quality of service.
- The secondary user may not achieve the required quality of service even when sharing the licensed spectrum band.
- Some primary networks refuse to lease a portion of its spectrum to cognitive radio networks, on the fear of affecting their users' communications.
- Complex protocols and protocol layers design must be developed to perform the cognitive radio networks' functions mentioned in sections 2.4 and 2.5.
- Additional infrastructure and signaling overhead are required.

- They have to deal with several secondary users and entities from several networks with different infrastructure and requirements.
- Increased competition as the cognitive radio networks applications, used fields and researches increases.

## **2.8 Summary**

This chapter overviews cognitive radio network; its main concept, definition, components, classification, main functions and their involvements in the cross layer design, its main applications and finally its main advantages and disadvantages.

As this thesis addresses the problem of spectrum sharing in Cognitive Radio Internet of Things (CRIoT); the next chapter overviews Internet of Things.

# **Chapter 3**

## **Internet of Things**

# Chapter 3

## Internet of Things

The term the Internet of Things (IoT) has been attracting several research concerns. Several IoT applications have been developing, which leads to developing various technologies and research disciplines to support deploying IoT applications.

IoT is defined as a worldwide network of interconnected objects/things/devices which allow devices that can be assigned an IP address to transmit data over a network. IoT devices can be connected either in a wired or wireless manner, however wireless connection is frequently used due to its flexibility. The desired communication is with minimum human intervention, only the devices will interact together [43],[44].

This chapter gives an explanation about the IoT concept in general and the fog computing that was introduced to support the IoT applications, it is divided into six sections. The first section gives a comprehensive IoT definition. The second section illustrates the IoT architecture layers functions. The third section outlines the most common IoT communication technologies. The most famous IoT applications are overviewed in section four. Fog computing is defined in the fifth section, together with its usage for IoT. The last section summarizes the chapter.

### 3.1 IoT Definition

Internet of Things (IoT) definition has been evolving during the last few years due to the evolution and the development of the technologies contributing for establishing a network of smart devices, so several definitions were made to describe the IoT deeply.

The authors in [45] believed that the IoT's aim is to be involved and affect each aspect in our life, so they stated two definitions for IoT. The first definition is as follows: IoT is a network of things, these things can be several items such as devices, vehicles, etc. An IoT network's device is embedded with electronics, software, sensors, it can be assigned an IP address to send or receive data over network. The

second definition is as follows: IoT is a network that gathers information from the real world, and controls the network devices through several deployed devices with network connectivity, that can perform perception, computation and execution.

The authors in [46] stated that IoT is a self-configuring dynamic network. It uses standard and interoperable communication protocols, physical network devices integrated with the network information through intelligent network interfaces. Also it can be defined as a recent technology that involves machine to machine, human to computer communications which will use every device that can serve the deployed application.

While the authors in [47] defined the IoT as a smart interaction between human and network devices for information reciprocation to be analyzed to form a new value. This is a complicated solution that leads to form three main technology components; network device connection with embedded sensors, network connectivity and final applications and information analysis.

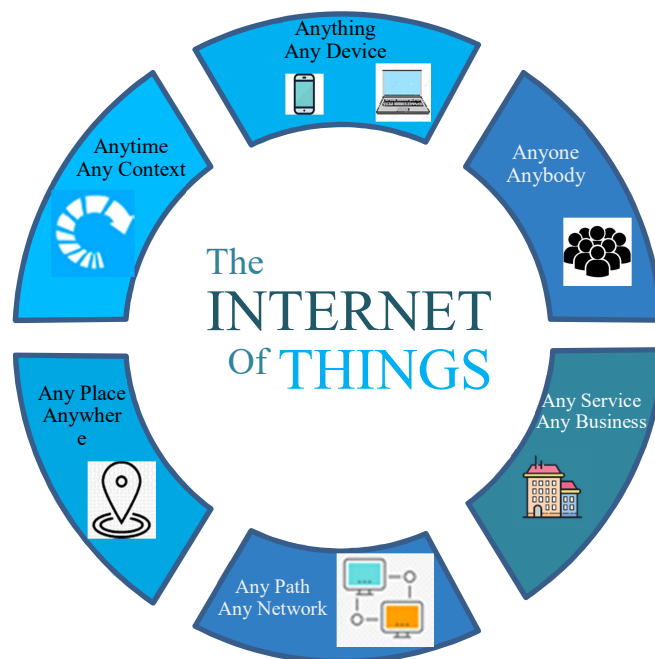


Figure 3.1 Internet of Things [48]



Also the authors in [49] says that IoT permit human and devices to be connected at anytime, anyplace, with anyone and anything, using any path or any network and any service. But finally the simplest definition is stated in [48]; IoT is a model of devices that can be sensors, actuators and processors to communicate with each serving a greater service.

### 3.2 Internet of Things Architecture

Till now there is no IoT architecture agreed upon globally, instead several architectures were proposed. At the beginning a three layered architecture was proposed, then some other researchers thought that the three layered architecture won't satisfy the applications requirements, so a four layered architecture was introduced. However due to IoT development another five layered architecture was proposed to fulfill the IoT security and privacy requirements. The development of the IoT architecture is shown in figure 3.2 [50], [51].

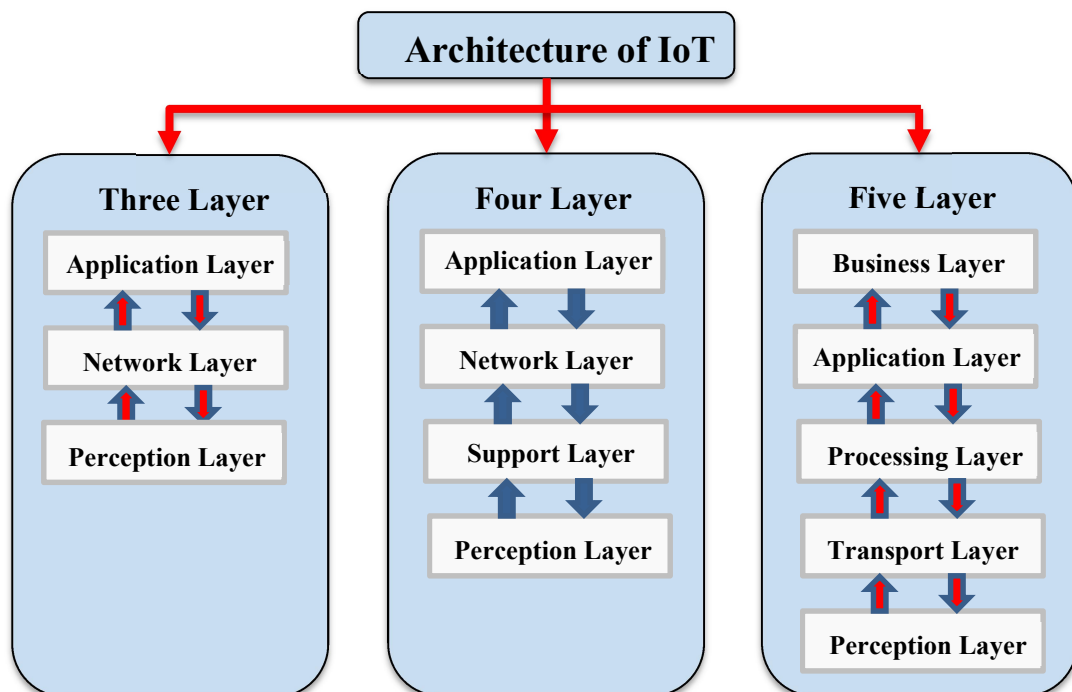


Figure 3.2 Three, Four and Five Layered IoT Architecture [50]

### 3.2.1 Three Layered Architecture

This is labeled to be the most basic IoT architecture, which was introduced in the early stage of the IoT research that only accomplishes the basic IoT idea. The three layers are the perception layer, the network layer and finally the application layer as shown in figure 3.2.

- **Perception Layer:** It is the physical layer or as sometimes called the sensor layer. This layer consists of sensors chosen according to the deployed application. The sensors gather information and identify devices in the surrounding environment.
- **Network Layer:** It is also known as the transmission layer. This layer is responsible for the communication, and for transmission of the gathered information for the lower layer to other smart devices, network devices and network servers.
- **Application Layer:** Having several IoT applications, this layer is responsible for providing the user with the deployed IoT application services.

### 3.2.2 Four Layered Architecture

As the three layered architecture was quite basic, it was found that it didn't satisfy all the IoT requirements, so the field researchers proposed the four layered architecture. These four layers are the perception layer, the support layer, the network layer and finally the application layer as shown in figure 3.2. The functions of the three layers common between the four and the three layered architecture are the same.

- **Support layer:** This layer was added for security reasons, it has two functionalities. It confirms that information is transmitted by authentic users and preserved from threats. Also it sends reliable information to the above network layer.

### 3.2.3 Five Layered Architecture

Although the four layered architecture contributed significantly in the IoT architecture development, but there were still some missing issues concerning security and storage. This leads to the five layers architecture; these layers (shown in

figure 3.2) are the perception layer, the transport layer, the processing layer, the application layer and finally the business layer. The perception and the application layers have the same role as in the three layered architecture.

- **Transport layer:** This layer transfers the sensed data received from the perception layer to the processing layer through the network and vice versa.
- **Processing layer:** It is also known as the middleware layer. This layer collects, processes and stores the information sent from the transport layer to eliminate the useless information and extract the useful one. It deploys several technologies such as big data processing, databases and cloud computing modules.
- **Business layer:** This layer manages and controls the entire IoT system through managing and controlling the IoT applications, business, profit models and user's privacy. It also defines how the information is generated, conserved and modified.

### 3.3 Internet of Things Communication Technologies

Several communication technologies exist in our real world, these communication technologies can be used for deploying the IoT application. Selecting the used communication technologies is mainly based on the applied IoT's application. Due to the special properties of most of the IoT devices, the used IoT communication technologies must have several requirements; low cost, low processing capacity, low power, low bit rate and low storage capacity. IoT communication technologies can be classified into two main categories; short range communication technologies and long range communication technologies, the most significant ones are explained below [48], [51]–[53].

#### 3.3.1 Short Range Communication Technologies

In the following we present the most common short range communication technologies used for IoT devices.

- **Radio Frequency Identification (RFID):** Using the electromagnetic fields to transmit data and automatically identify and track it through RFID tags attached to the devices. It is suitable to be used for patient tracking in a healthcare facility.

- **WiFi:** It is based on IEEE 802.11 communication protocol family standard. WiFi HaLow is a low power WiFi which is based on IEEE 802.11ah. It is better than the classical WiFi as is it characterized to be lower power and have larger communication technology. It can be considered a key technology for IoT.
- **Zigbee:** It is based on IEEE 802.15.4 communication protocol standard used for personal area network. It has a fixed protocol design. It is characterized to be low cost, low data rate, relatively short transmission range, scalability, reliability, and flexible protocol design communication technology. It is mostly used in home applications, agriculture, industry monitoring and medical observations.
- **Bluetooth:** It is a very low power communication technology used to transfer data between short range devices. It is characterized to be short range, low power consumption communication technology. It is suitable to be used by wearable electronics devices, such as headphones, sensors, etc.

### 3.3.2 Long Range Communication Technologies

In the following we present the most common short range communication technologies used for IoT devices.

- **Sigfox:** It is a low power wide area network that is an alternative network operator. It connects low power devices. It is characterized to be long range, low cost and low data rate communication technology.
- **Long Range Wide Area Network (LoRa WAN):** It is a low power wide area network. It is similar to Sigfox protocol. It allows low power devices to transfer over long range wireless connections. It is characterized to be long range and low power wireless platform.
- **Weightless:** It is a low cost technology to be integrated into machines. It is characterized to be low cost and low power communication technology. It can support encryption and mobility. It also has the ability to handle a large number of devices. It is suitable to be used in vehicular tracking, smart meters health monitoring and industry observations.
- **Narrow band IoT (NB-IoT):** It is based on narrow band communications. It was mainly constructed to handle large number energy constrained devices. It is

characterized to be low power and low cost communication technology. However its main problem that it doesn't support handoff.

- **Long Term Evolution (LTE-M):** It is also known as enhanced Machine Type Communication (eMTC). It is a low power wide area network, which was introduced for sending small amount of data over long period of time. It is characterized to be low power and low cost communication technology. The used devices can have SIM card to communicate through cellular networks. It suitable to be used for alarm systems and smart meters.

### 3.4 Internet of Things Applications

The IoT technology can be emerged in several applications in our real life. IoT has evolved rapidly so it can be applied in many application domains such as domestic applications, societal applications, environmental applications, technological applications and emergency and critical situations as shown in figure 3.3. Each of these domains is a research umbrella of IoT applications. Domestic application is the applications related to home, household, household affairs. It can include smart living rooms, smart washroom, smart kitchen, smart cooling systems, smart doors and elderly supports, etc. Societal application is related to emerging the IoT devices into cities to gather data to be used later to improve infrastructure, public utilities and services, etc. This type of applications can include smart city, smart transportation, smart parking, smart marketing, smart traffic, and smart healthcare. As for the environmental applications IoT devices are installed to detect and measure environmental changes, it can include smart farming, smart wildlife, smart ecosystems and smart water, etc. The technological applications include intelligent shopping and smart security, etc. Emergency and critical situations applications are a crucial domain that may include natural disaster, radiation and water leveling, enemy intrusions and unstable industries, etc.

In the following subsection some of these IoT applications will be explained [45], [47], [59], [48], [51], [52], [54]–[58].

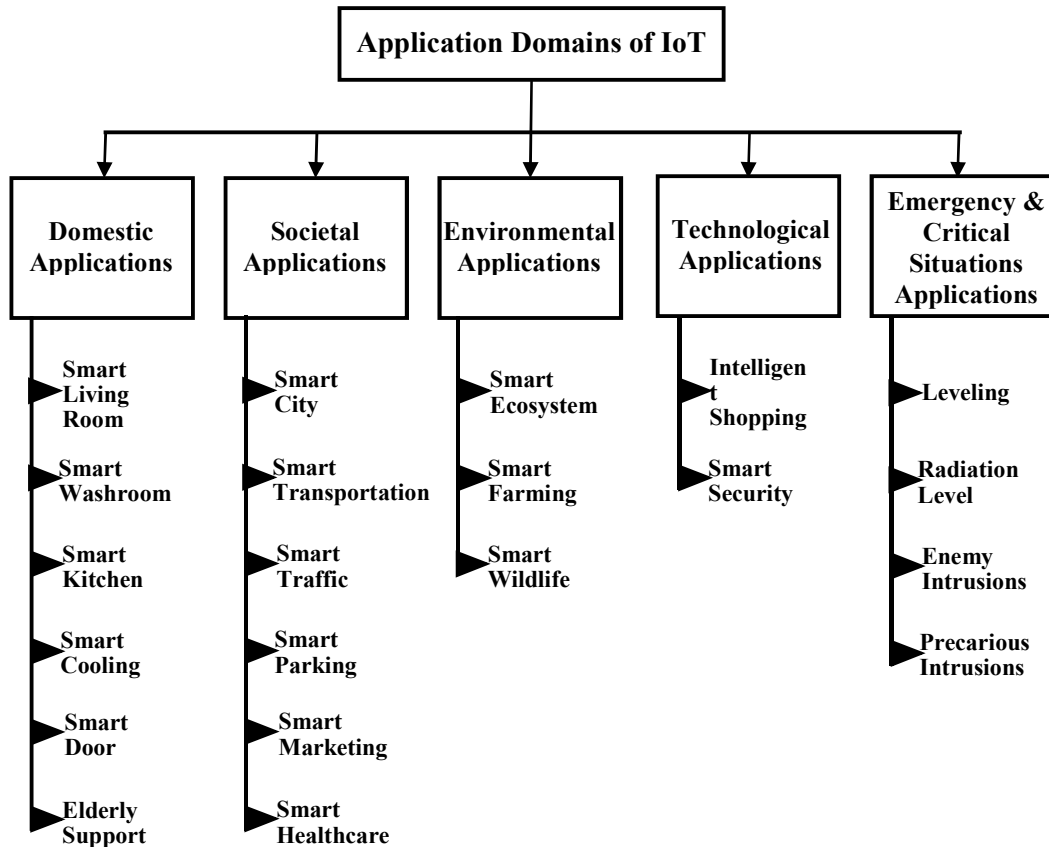


Figure 3.3 IoT Applications Domains [45]

### 3.4.1 Smart Home and Offices Applications

Smart homes can be considered as automated buildings that can contain several applications such as smart living rooms, smart washroom, smart kitchen, smart cooling, smart door, smart media, smart lighting, elderly support, security, etc. In these applications; sensors and actuators are deployed to make our environment more comfortable. Also it is useful to manage and observe the processes taking place indoors such as lighting, temperature and music. Smart home application is also used to monitor, utilize and save energy and water consumption. Furthermore it is useful for intrusion detection to prevent from any intrusion by detecting the doors and windows conditions.

### 3.4.2 Smart City

Smart city can include several IoT applications, where numerous things such as sensors, actuators, lights and meters are installed in the environment to gather data to

be analyzed. The analyzed data is used to serve several purposes such as improving citizen safety, improving city maintenances, monitoring the buildings, roads and the bridges physical conditions. The smart city IoT applications improve several things such as the citizen and the government engagement, improve the city infrastructure, improve the public utilities and improve the citizen life quality.

### **3.4.3 Transportation Applications**

Smart transportation, smart parking, smart traffic control and fuel consumption can be considered as IoT transportation applications. Smart transportation includes several research areas; transportation analytic, transportation control and vehicle connectivity. Smart devices such as sensors and actuators are either installed at the side road or embedded into the devices (vehicles, traffic lights, mobile phones, etc.), they are connected together through networks as shown in figure 3.4. These things collect and transfer information to be analyzed, in order to be used for several issues such as predicting the vehicles or people that will be using a transportation facility. It also allows monitoring the entire city traffic management, by monitoring the routes, vehicles and pedestrians to reduce the road congestion, and the time people spent on the road. One of the most important roles in smart trafficking is accident avoidance which can save several lives. Monitoring the availability of parking spaces in the city to direct people directly toward the available spaces can save time, fuel and reduce traffic flow. Smart traffic lights can be used in emergency cases such as passing ambulance or police car. Also sensors can be embedded into vehicles to gather information about the amount carbon dioxide in the atmosphere to be used later for global purposes.

Note that several researches list the transportation application under the smart city IoT applications, however here it is better to be separated as the IoT transportation application is a big research area, it was lately considered as a hotspot research area in the IoT applications research areas under which several researches was proposed.

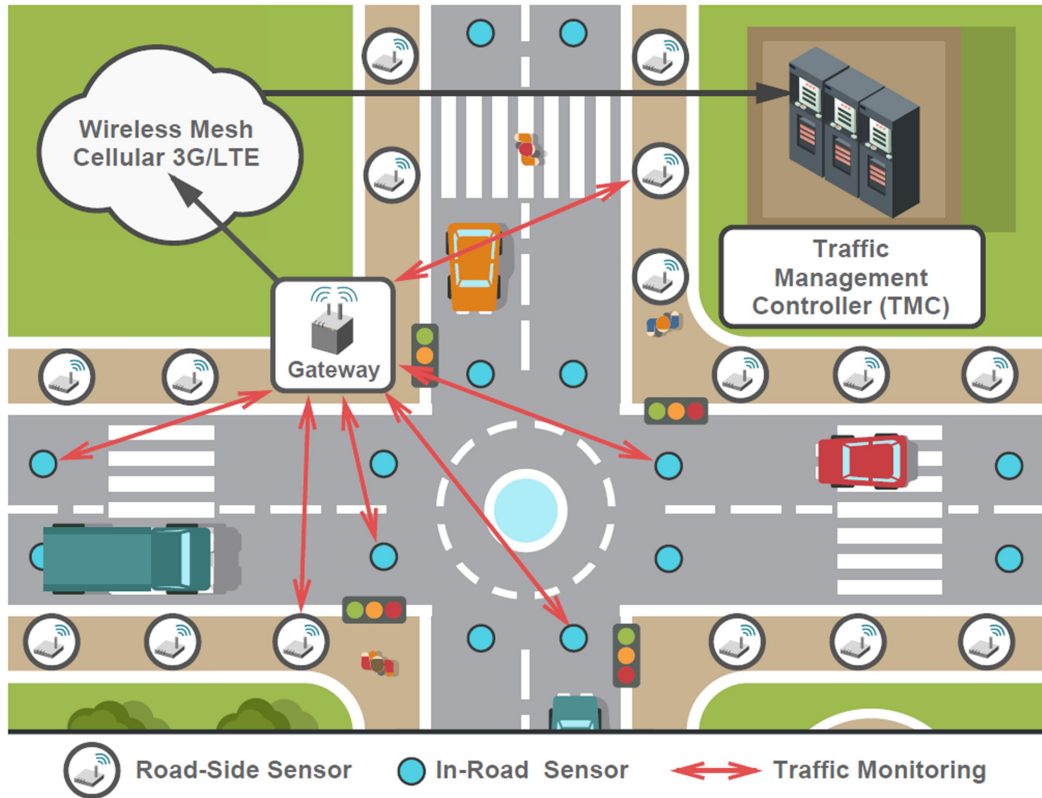


Figure 3.4 IoT transportation Application Example [60]

### 3.4.4 Healthcare

Several researches were done considering this application, as IoT was considered a suitable technology to develop a healthcare or eldercare facility, which introduces novel methods to monitor and treat several diseases as shown in figure 3.5. Wearable devices sensors are used to monitor and record the patient/elder health conditions by continuously sensing the body vital signs such as heart rate, pulse, digestive system, and blood pressure to transmit warning in case of any abnormality. Also sleeping patterns and medication schedules can be monitored. Furthermore the collected data can give another perspective to symptoms and trends. It allows patients to have more knowledge and control over their case and treatments, and allow practitioners to access their patient health information remotely. IoT tagged sensor devices can be embedded with the medical devices such as wheelchairs, defibrillators, nebulizers, oxygen pumps and other monitoring equipment to develop real time tracking for them.



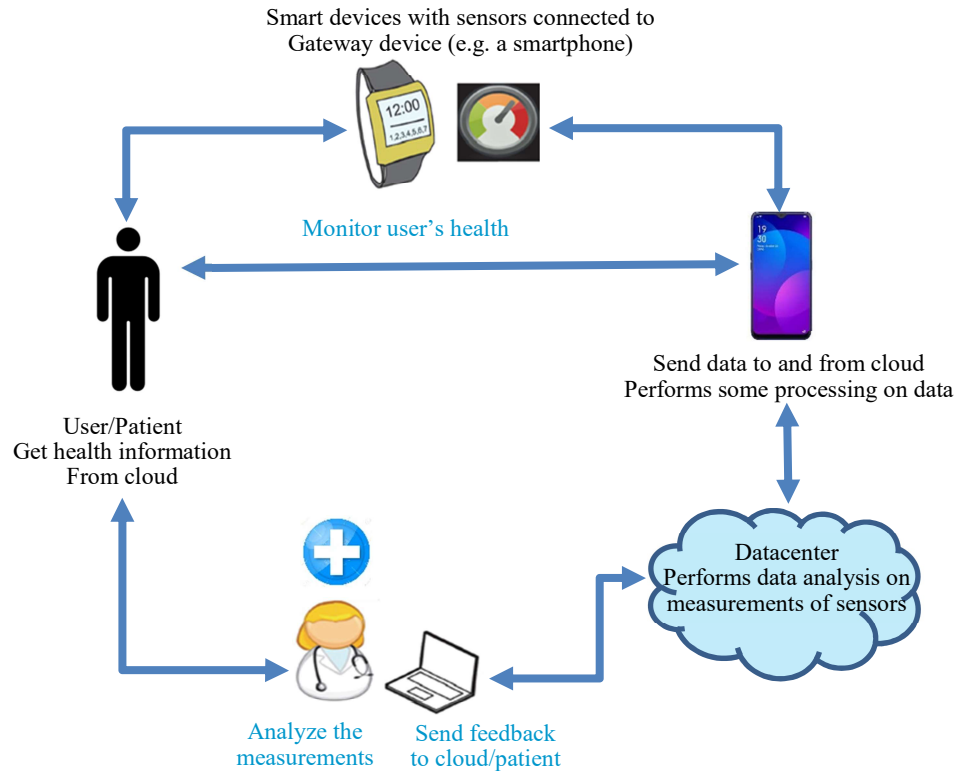


Figure 3.5 Block Diagram of a Smart Healthcare System [51]

### 3.4.5 Smart Grid

To allow modern electric power generation, transmission, distribution and consumption regulation, smart grid IoT is deployed to add some intelligence to each stage as shown in figure 3.6. This allows two way power flow (from supplier to consumers and vice versa), which can save energy and help consumers to understand the power flow and the dynamic pricing concept. Smart grid is defined as a data communication network where sensors are integrated with the power grid in order to gather data about transmission lines, distribution stations and substations and consumers. It can be considered as the firm resolution for future grid managements. The collected data is analyzed to improve the reliability, efficiency and electricity economics. It also influences customer service and bills accuracies. It saves energy by reducing the consumption and can help in detecting the frauds and technical losses in the grid. Likewise it can help in leveling the energy consumption curve by reducing the curve peak.

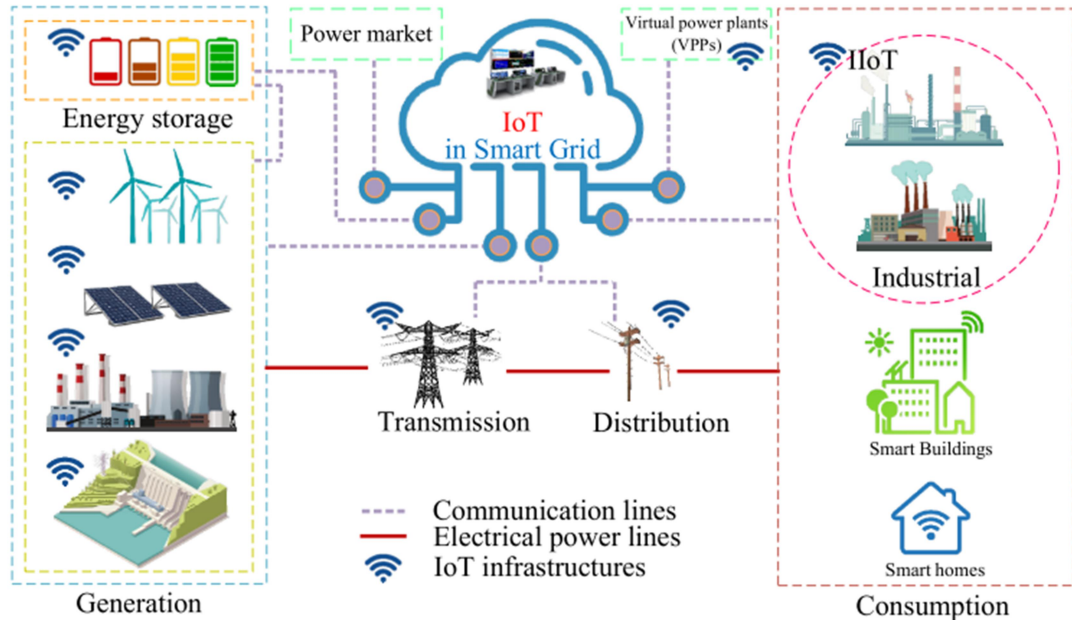


Figure 3.6 IoT Smart Grid Application [61]

### 3.4.6 Smart Agriculture

Smart agriculture in IoT is considered to be continuously growing, where smart things are installed to be used by the farmers to measure several weather parameters such as light, humidity, temperature, soil moisture, etc. These measurements are done in real time, and then sent to servers to be analyzed as shown in figure 3.7. This IoT application is not only used to monitor the climate conditions, but also to monitor the location and the health condition of the farms' livestock such as cows, hen, goats, etc. Also the crop can be remotely monitored in term of sample size, time, location and the amount of residues, to maintain the crop quality. Likewise the green house automation is considered as a great IoT agriculture application. All of these services help the development of a smart farm management system that helps the farmers to understand more about the future farm and the crop situation, which may reduce the loss risks.

Several more IoT application has been proposed and researchers are developing more every day, we have only mentioned the significant and most researchable ones.

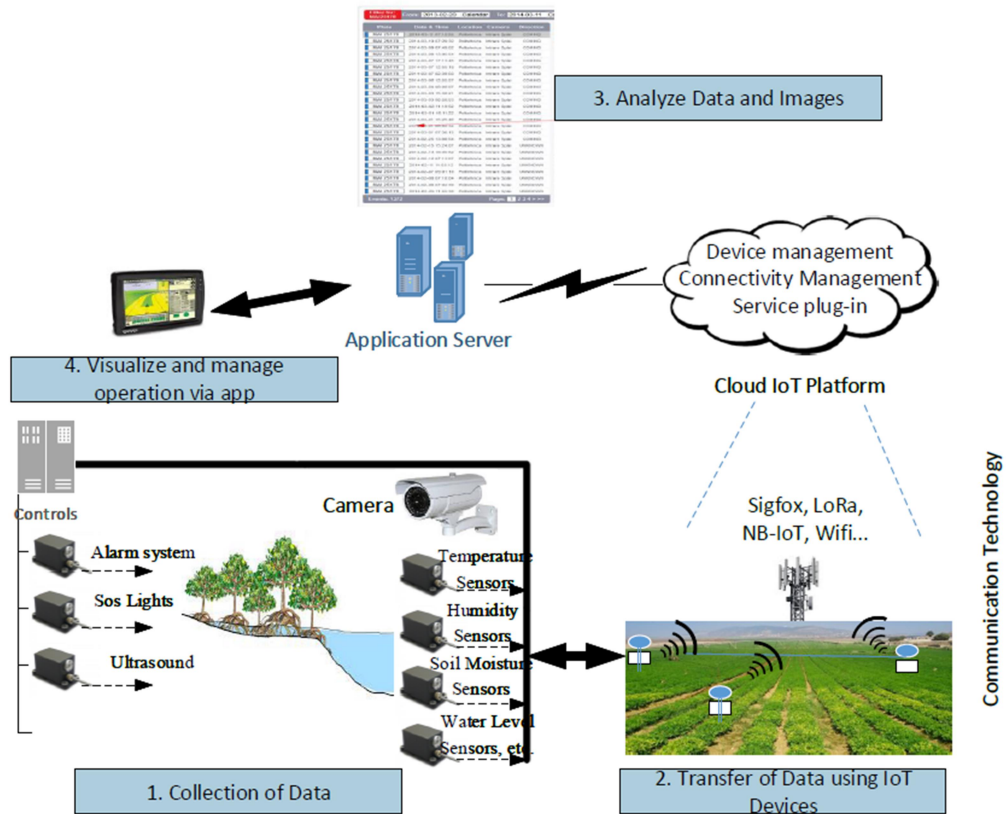


Figure 3.7 IoT Agriculture Application [54]

### 3.5 Fog Computing

IoT applications are attracting more attention in several aspects and have been developing rapidly. So the IoT platform will be dealing with huge number of devices and most of these devices are very basic, that means they won't be equipped with neither compute nor data storage facilities.

Figure 3.8 shows the most significant requirements required when designing and developing an adaptable and scalable IoT platform [62]–[65]. Consequently external resources are required to perform a significant part of the required data storage and processing.

Cloud computing was the first thought to be used for the data processing and storage of the IoT application, but unfortunately it was considered an unfeasible solution for numerous reasons such as; the far location of the cloud servers, which leads to high latency that some IoT services can't tolerate, the huge bandwidth

required to transfer the data back and forth the cloud serves also act as an issue when using the cloud computing. Furthermore developing a secure centralized data center to act as general and multipurpose platform is neither a feasible nor a practical solution. So fog computing was introduced to develop locally computing, processing and storage facilities at the edge of the network. Fog computing acts as an extension of the cloud computing located nearby the IoT application devices [62], [63].

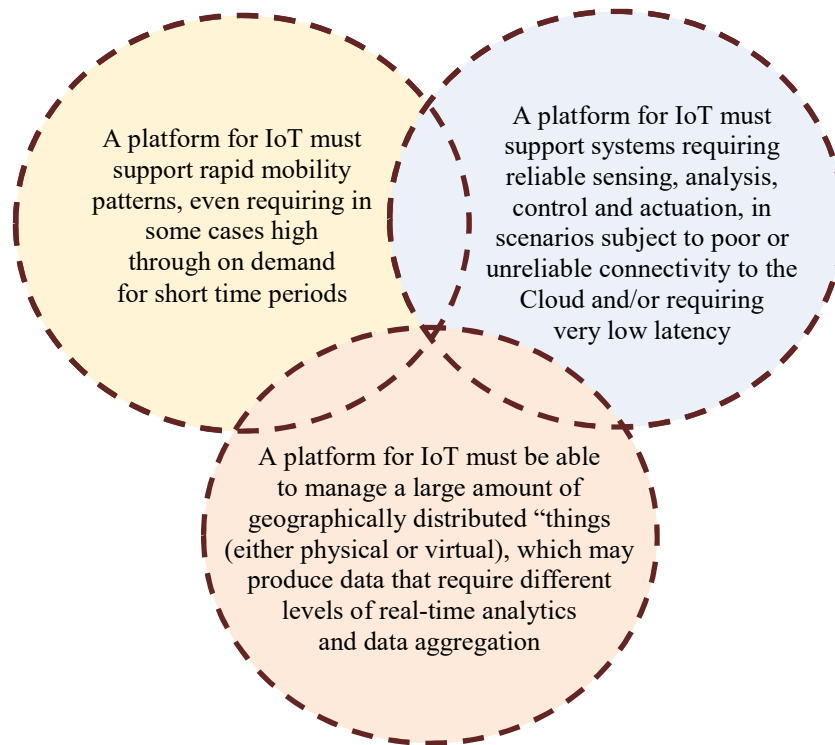


Figure 3.8 IoT Platform Main Requirements [63]

Fog computing introduces low latency and devices location awareness, real time decision making, network bandwidth utilization services compared to cloud computing. The distributed architecture of fog computing secures the offered services in a better way than using cloud. Of course, IoT platform won't depend only on the fog node and dispense the cloud usage, however processing and computation, networking, storage and capabilities will be hierarchical from the cloud till the fog nodes at the edge of the network. The number of hierarchical levels depends on the application the IoT is deploying for and the network infrastructure. The fog computing architecture is a system hieratical layers architecture as shown in figure 3.9. The hieratical layers architecture's purpose is to distribute the computing,

processing, networking and storage to provide capacity balance among the hierarchical layers. The fog computing layer(s) is located between the cloud layer and the smart devices. Each of these layers has different properties with respect to the storage's capacity, the network bandwidth, the processing and computing capabilities, the response time, etc. [64].

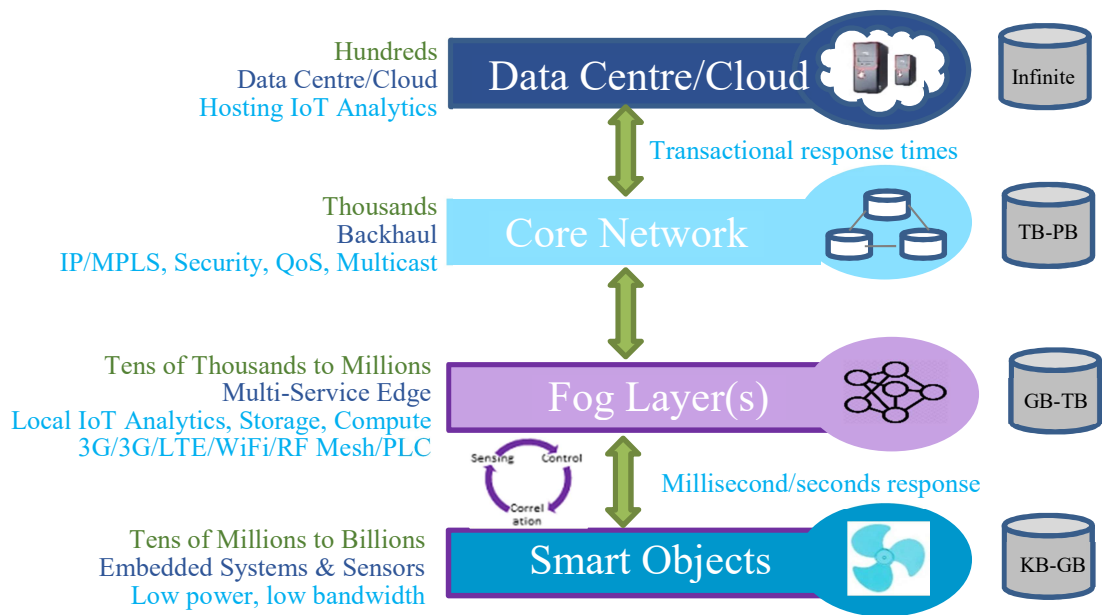


Figure 3.9 Fog Computing Architecture [64]

### 3.6 Summary

This chapter gives a brief overview about Internet of Things (IoT). First IoT is defined. Then, the IoT architecture layers functions are explained. After that the most common IoT communication technologies and the most famous IoT applications are illustrated. Finally Fog computing is defined together with its usage for IoT.

As this thesis addresses the problem of spectrum sharing in Cognitive Radio Internet of Things (CRIOts), the next chapter main objective is to overview the existing CRIOts spectrum assignment (sharing/allocation) protocols.

**Chapter 4**

**Spectrum Sharing/Allocation**

**for**

**Cognitive Radio Networks**

**Internet of Things**

# **Chapter 4**

## **Spectrum Sharing/Allocation for Cognitive Radio Networks Internet of Things**

Many IoT applications have been developed recently, which leads to the increase of the network load. To avoid the lack of spectrum, for IoT communication, it is recommended to use cognitive radio networks [43], [44]. This chapter focuses on Cognitive Radio Internet of Things (CRIoTs). It overviews most of the researches addressing channel allocation and packets scheduling, when merging the cognitive radio networks with the IoT technology. The researches done in this area shows that this merging process is still in its early stage, and needs a lot of developments.

The chapter is divided into four sections. The first one defines CRIoT and illustrates its design factors. Section 2 reviews the common challenges for CRIoTs. Existing CRIoTs spectrum assignment (sharing/allocation) protocols are surveyed in section 3. Section 4 summarizes the work presented in this chapter.

### **4.1 Cognitive Radio Internet of Things (CRIoTs)**

The continuous growth of IoT technologies and applications, leads to massive amount of data needed to be sent on the spectrum band, which we already suffer from its scarcity. Spectrum scarcity not only depends on the channel availability but also depends on the spectrum utilization and the used technologies. The ordinary fixed spectrum assignment policy became inadequate. A simple solution is to purchase the spectrum, but this will be costly.

Due to the dynamic conditions of the environment, the spectrum allocation and utilization and the cost of purchasing the spectrum, it is envisioned that IoT won't be able to grow as estimated. Merging the IoT technology and the CR abilities can help in the IoT spectrum scarcity problem. In other words, the IoT will suffer from lack of

spectrum as the number of IoT devices will increase, and there will not be enough spectrum allocated for it. The IoT network needs to have cognitive radio capabilities, so the IoT devices will be able to use the licensed unutilized spectrum band. Providing opportunistic access of the spectrum bands to the IoT devices, will increase the spectrum efficiency.

The usage of CRNs to establish the IoT smart network will be an efficient and economical solution to the IoT spectrum scarcity problem. CR technology will be merged with the IoT to be called Cognitive Radio Internet of Things (CRIoTs). CRIoTs will be used in many application especially the time sensitive ones such as smart health care and smart transportation. The merged technologies should satisfy many network parameters such as channel availability, allocation delay, end-to-end delay, reliability, energy efficiency, high throughput, etc. Many communication standards and technologies have to be used to cope with the increase of connectivity among the numerous number of IoT applications and services that use the CR technology. For the indoor smart environment Bluetooth, ZigBee and WiFi, etc. are used, and for the outdoor smart environment cellular system, IEEE 802.11af, Weightless, etc. are used. With CRNs capabilities both short and long range communication will be supported [44].

The IoT smart networks have to perform several functions. It will capture the interference free channels with the presence of PU. Estimate the quality of service of the detected holes. Provide continuous communication in case of any change in the operating frequencies, either for the arrival of a PU or due to its mobility. And finally regulate the licensed spectrum access for the huge number of IoT devices with minimal interfering with the PUs [44].

Figure 4.1 illustrates the design factors for CRIoTs system and their relation to each other's [66]. The first factor is the application in which the CRIoTs system is used and consequently its characteristics. The type of used application determines the nature of the signal to be transmitted whether analog or digital, data size, transmission type, spectrum range, and other deployed standards. The second factor is the technology used for communication between devices and its characteristics. IoT deploy several technologies to allow IoT devices to communicate, those technologies specifies several things such as data rate, transmission range, operating



frequency and the standard MAC protocols used to fit the application type. The last one is the regulation imposed by each of the countries using the system such as operating band assignment for both licensed and unlicensed spectrum, allowable interference and certificates.

Note that Cognitive Radio Internet of Things (CRIoT) is merging the cognitive radio capabilities with the IoT technology. So that the IoT can use the licensed spectrum to enhance spectrum scarcity explained previously. This is different from Cognitive Internet of Things (CIoT) where there must be devices that are equipped with cognition capabilities which are to learn, think, and make intelligent or/and critical decisions by comprehend of the surrounding worlds [44][67].

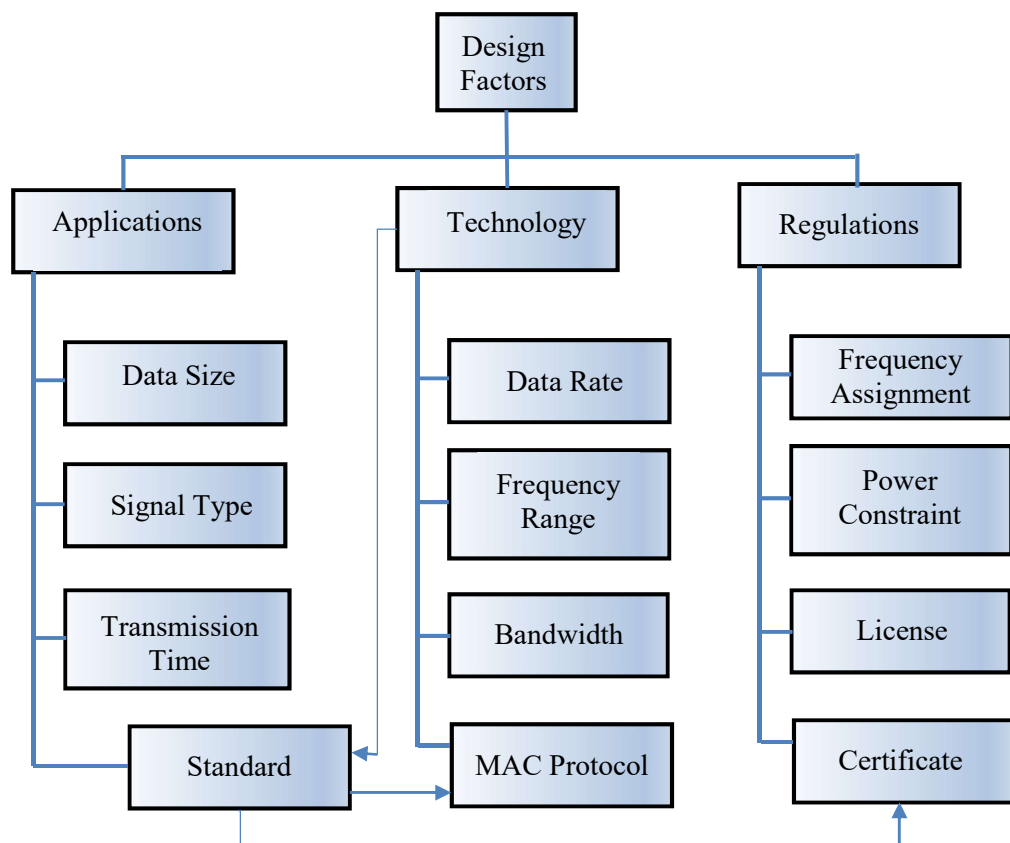


Figure 4.1 Design Factors of Cognitive Radio for Internet of Things System [66]

## 4.2 Cognitive Radio Internet of Things Challenges

Although various challenges for CRIOts depend on the designed application scenario and the existing network structure, there exist common challenges for all Cognitive Radio Internet of Things [68], [69], these challenges are considered as open research areas. These common challenges are:

- **Standardization;** Existence of multiple technologies is considered as one of the major challenges in the development of IoT applications. Note that IEEE 802.22 is a standard for wireless regional area network that uses the white spaces in (TV) frequency spectrum. It can be considered as a step for CRNs standardization. Now, there are great efforts for standardization of CRIOts.
- **Mobility;** CRIOts system must work perfectly in spite of the IoT devices mobility, which leads to the change of the topology and of the free holes to be used.
- **Resources availability;** Most IoT applications have a huge number of IoT devices over the same network, this requires considerable network resources to be enough to support their operations.
- **Energy consumption;** CRIOts concept will consume considerable electrical energy especially in spectrum sensing phase, so energy consumption has to be as minimum as possible. Prediction of the empty spectrum spaces can be considered as a good solution if done efficiently. Also energy harvesting must be taken into consideration to be able to provide sufficient electrical energy to reach the intended goal.
- **Scalability;** IoT applications will connect million or even billions of IoT devices over the same smart network. A channel allocation protocol for this number is not an easy task. Extendable services and operations have to be developed to tolerate new services and devices joining the network constantly.
- **Coexistence of Heterogeneous Communication Devices or Interoperability;** The existence of heterogeneous devices and protocols that need to operate with each other, so the system needs to work with several heterogeneous devices, standards, technologies and protocols.

- **Reliability;** For IoT devices to access the licensed spectrum dynamically, changing some transmission parameters may be performed such as transmission power, modulation mode, channel selection, antenna parameters, etc. Therefore, the spectrum allocation function will require multiple variable optimization process, which will increase the complexity of the allocation process. Also, the IoT devices may have different quality of service requirements such as delay constraint, energy consuming rate, packets dropping rate constrains, etc. Because of the IoT time sensitive applications and the numerous data resulted from IoT devices, the required spectrum allocation must be fast and complex in order to meet several objectives.
- **Distributed networks types management;** IoT network is expected to be composed of a significant number of IoT devices existing in various types of networks with various topologies and specifications. These networks will have different communications techniques, configurations, failures and performance. They must negotiate with each other and be self-organized to share the network resources effectively.
- **Security;** Hackers can launch attacks and infiltrate IoT networks to control devices to control the IoT devices, such as channel information, energy consumption, or any other resources, so high security will be required to ensure integrity of the transmitted data. Note that different levels of security will be required based on the type of data, which leads to high complexity.

As mentioned, each point in these challenges is be treated as a considerable open research area, our research is concerned with the resources availability, scalability and reliability challenges.

## 4.3 Cognitive Radio Internet of Things Spectrum

### Assignment Survey

As shown in figure 4.2, we classified the spectrum assignment protocols for CRIoTs into two types, spectrum sharing and spectrum allocation. This classification was done based on the existence of cooperation between primary and secondary networks. Few researches discussed the problem of CRIoTs spectrum assignment,

this section surveys all these protocols (to the best of our knowledge) that addressed this problem. This section is divided into two subsection; spectrum sharing and spectrum allocation.

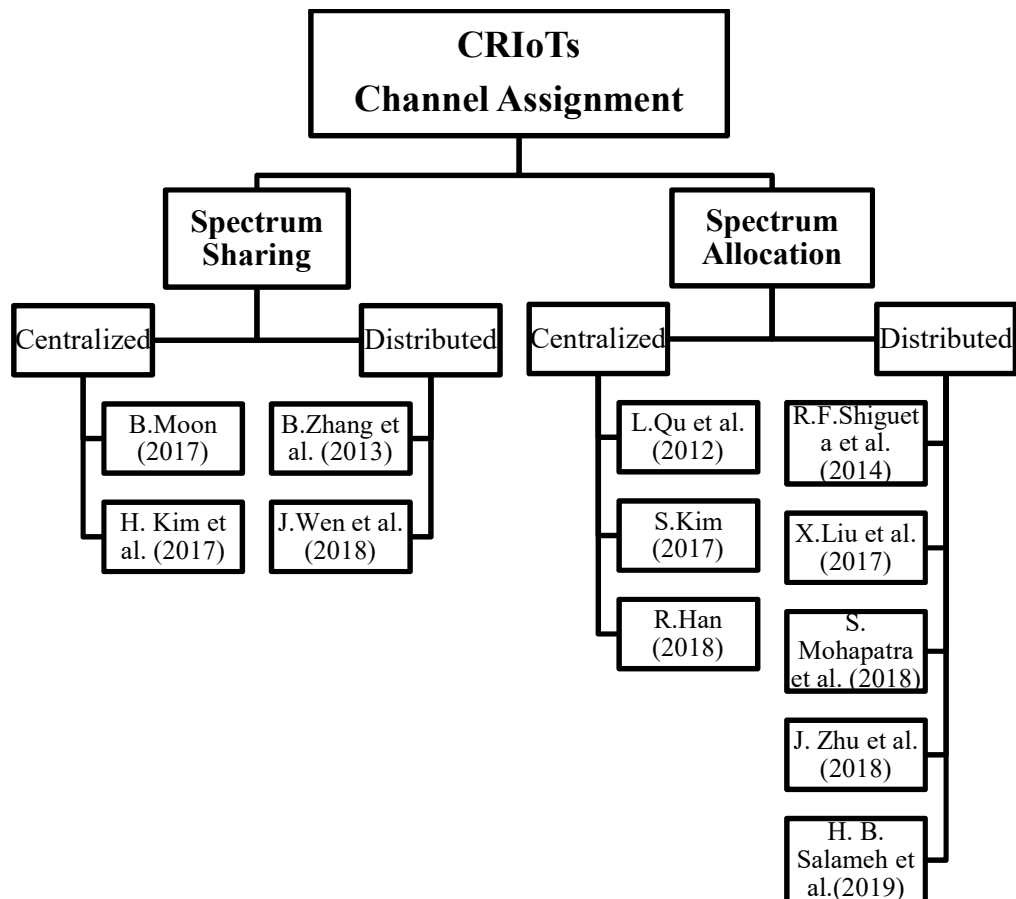


Figure 4.2 Classification of Channel Assignment for CRIOts

### 4.3.1 Spectrum Sharing

In spectrum sharing there is cooperation between both the primary users and the secondary users (IoT devices). In other words, the primary network(s) leases its free slots to IoT devices, of course without degradation of the PUs quality of service. This section surveys these protocols.

Authors in [70] proposed a cooperative transmission model between the PUs and the SUs. It leases the spectrum from the primary network to the SUs, in order to enhance the sum of utilities for both the PUs and SUs. Game theory is used to solve the resource sharing problem. Nash Bargaining Solution (NBS) and selective

cooperation are used for acquiring the intended optimal leasing strategy. The problem is modeled as a cooperative two-player bargaining game between the PU and the SUs.

It begins when a PU broadcasts a Request-to-Send (RTS) to inform its Primary Receiver (PR) with its minimum transmission rate. The PR then anticipates the channel's condition and calculates the transmission rate. If the channel meets the required transmission rate, the PR replies by a Clear-to-Send (CTS). Both RTS and CTS are received by all SUs. If the transmission rate of the PU is less than the minimum transmission rate, this indicates that the PU will cooperate with a SU. Each of the SUs will calculate the bargaining game parameters and the best strategy, this leads to selecting the best optimization which can maximize the sum of utility for both the PU and the SU. Note that the PU utility is the sum of the gains from both the transmission rate changing and power saving, while that of the SU is the achieved transmission rate and the energy consumption. After that each SU sets a timer, the one that its timer runs out first will send a Partner-to-Send (PTS) that includes the best cooperative strategy between this pair. Finally, when the PU and the other SUs receive the PTS message, that SU will be chosen to share the channel with that PU. Simulation results showed that leasing the spectrum while using cooperative game theory will enhance the sum of utilities for both the PUs and the SUs. The main disadvantages of this method are: first, the performed processes are too complicated to be done by an IoT device due to its limited capability. Second, the number of IoT devices used to evaluate the protocol is too small. Third, the performance of the proposed protocol is not compared with any other protocol.

Authors in [71] proposed a dynamic spectrum access technique for IoT devices using cognitive radio-enabled low power wide area networks, running in both the unlicensed and licensed band. The main objective is to maximize the unlicensed users' spectrum capacity, while never interfering with the licensed Low Power Wide Area Networks (LPWANs) primary users. The authors supposed that the entire spectrum is divided into sub bands. Each PU can use an A-band channel including several sub bands. Each SU can use a B-band channel which consists of only one sub band. The A-band and B-band channels overlap with each other. The PUs have the advantage to operate on the overlapped band. The existence of SUs is completely

unknown to the PUs. Therefore, if a PU aimed to use a channel in the overlapped band while the channel is occupied by a SU, the SU has to instantly jump to other available unlicensed channels. If there is no other empty channel, the SU fails to change its operating channel, they are queued at the top of a SUs queue. When the number of PUs running-calls reached its maximum or there are no enough free channels and a PU call is needed, it will be blocked and cleared. The SUs' requests queued are served as a First Come First Served (FCFS).

The authors derived an equation to calculate a cutoff value to determine the ratio of the overlapped area that is divided between the PUs and the SUs. This value has to be adapted carefully so an adequate tradeoff between exaggerated delay experienced by the SUs calls and availability of further spectrum must be done. Simulations and numerical analysis are used to evaluate the proposed strategy. The results proved that the proposed technique maximized the spectrum capacity for the SUs while keeping the quality of service for the PUs. The main disadvantages of this strategy are as follows: first, blocking of PU calls is an option. Second, as the channel holding time for PUs calls increases, the average dwell time for SUs calls increases. Third, in case of high PU load, the queue size may grow infinitely. All of these disadvantages prevent let the protocol from dealing with real time packets. Finally, the performance of the proposed protocol is not compared with any other protocol.

In [72] a distributed cognitive random access algorithm was proposed. As it is impractical for a mobile IoT device to sense the spectrum, the authors supposed that the CRIOts network has wired spectrum sensors that various mobile IoT devices communicate with. As the distances between the IoT devices and the sensors are variant, there exist observation errors of spectrum occupancy. So the authors proposed that the IoT devices should adjust their probability of transmission based on the measured interference by the sensor. The aim is to improve the spectral usage and the sum of IoT devices data rates in the area, while satisfying the primary users' requirements. Stochastic geometry was used to derive a conditional interference distribution at each IoT device using the level of interference obtained from the sensor. Using the calculated conditional interference distribution, the optimal

transmission probabilities for all the IoT devices were calculated, and then a simple algorithm was derived to assign each IoT device its corresponding probability.

To evaluate the proposed algorithm (named area spectral efficiency), it was compared with conventional ALOHA scheme. Simulation results proved that using the proposed algorithm leads to better area spectral efficiency, and it is also more sensitive to IoT devices density. Its main disadvantage is that its architecture needs extra hardware devices. Also the Aloha protocol is a very old protocol to compare with.

The authors in [73] suggested the use of Secondary Information Gathering Stations (SIGSs) in order to collect the information for all IoT devices migrating in the area. The authors derived equations to be used in finding the optimal positions of the SIGSs and optimal operating channels for the secondary IoT devices. Particle swarm optimization is used to solve this problem. They used a fitness function that attempted to maximize the number of SUs and their capacity and conserving the primary network transmission, considering the Signal To Interference-Plus-Noise Ratio (SINR), the primary and secondary capacities and the appearance probability. Simulation results showed that the proposed algorithm works well in achieving the objectives. Compared to the Random Resource Allocation algorithm, it is found that the number of supportable IoT devices for the proposed algorithm is increased by about 20%. Its main disadvantages are, first its architecture needs extra hardware devices. Second, many assumptions are taken, such as the SUs already have information about the Primary users Base Stations (PBS) locations and the operating channel. Finally, they supposed that the PUs are controlled by a primary operator, so there is no possible interference from them.

### **4.3.2 Spectrum Allocation**

In spectrum allocation, the IoT devices (secondary users) network(s) are responsible for finding the free slots by themselves (through sensing) and use them. The sensing approach is in the scope of this thesis. Assuming that sensing was already done, this section surveys the protocols of how to use the sensed free slots.

Authors in [74] developed a spectrum allocation model based on Homo Equalis (HE) social model. It is a non-cooperative game model where the HE agents are the players predicting future behavioral strategy (accessing probability) in the next cycle, based on the collected information from various nodes. The distribution of the spectrum is done periodically among all nodes. Accessing probability is used to establish the nodes' next cycle behavioral strategies, in order to use the available detected holes fairly and effectively. For the players in the HE game, the desire to pursue imparity is inversely proportional with the good of playing. The allocation of the spectrum is done fairly by the antipathy characteristics of HE to inequality. Based on the recorded information (number of occupied and available channels, flow load, priority and communication time), the node's access probability is calculated. This probability will be used to establish the node's next cycle behavioral strategy with the HE community access schemes periodically, in order to use the available detected holes fairly and effectively. The proposed model consists of five components, each component leads to the following one to finally allocate the available channels fairly among the users. The five components are Radial Frequency (RF) receiver, RF perception, cognitive learning, decision and RF transmit.

Simulation results proved that this model improved the spectrum utilization. The main disadvantages of this model are, first its architecture needs added hardware devices. Second, having a single node channel leads to using the channel effectively but unfairly, and vice versa when using multichannel nodes. Third, the number of IoT devices used to evaluate it is too small. Fourth, the performance of the proposed protocol is not compared with any other protocol.

In [75] the authors presented a distributed spectrum allocation technique for IoT devices when merging the CR technology. This technique works in a distributed manner; it uses the traffic history for the next channel allocation. The technique used is divided into three main parts. The first part is the periodic distribution of Hello messages to inform the SUs of the free channels. The second part is the initial allocation algorithm, which is executed whenever a node is added to the network, or when the topology or a channel changes. First, the links leading to the added node is prioritized in a list based on their traffic history, and then each of these links is given a channel based on its order in the list. Based on the calculated priority of the node, it



determines whether it will be responsible for channel assignment or it will receive the assignment from another node. The third part is the interaction messages, which are sent by any node that changes its allocated channels, in order to inform the other nodes about that change. Upon receiving these messages, the nodes update their priority and run an algorithm to re-compute the channel assignment, taking into account the appearance of a primary user at any moment.

To evaluate this technique, its performance is compared to two older ones using simulation. Simulation results proved that this technique shows better performance in terms of aggregated throughput and packet delivery rate, but moderate performance for both reduced interference and end-to-end delay compared to the two other techniques. The main disadvantages of this method are, first the processes performed by each IoT device are too complicated, the IoT device may even be responsible to assign channels to others. This is disliked due to the IoT devices limited capability. Second, the number of IoT devices used to evaluate the protocol is small.

Game theory was used in [76], where a cooperative algorithm for both spectrum sensing and sharing in CRIoTs was proposed. Authors used an inspection game, with one inspector the cognitive radio base station, and multiple inspectees the IoT devices. The aim of the algorithm is to maximize the network throughput while maintaining the fairness among various IoT devices. Having the minimum and maximum sensing rate for the base station and the IoT devices, and the minimum and maximum spectrum slots needed for each IoT devices, the authors derived equations to calculate the following at each period of time: the utility of each IoT device and for the base station, the relative utilitarian bargaining solution also for each IoT device and for the cognitive base station, the aspiration level for each IoT device, the sensing rate update for each IoT device and finally the aspiration equilibrium.

The above calculated parameters are used to propose an algorithm for the IoT devices spectrum sensing and sharing. The spectrum sensing rate and allocation for each IoT device are randomly initialized. Then, the above parameters are calculated iteratively to change the values of spectrum sensing rate and allocation for each IoT device until the aspiration equilibrium value is satisfied. To evaluate the proposed

algorithm, a simulation model was built to compare its performance with that of the Adaptive Cooperative Spectrum Sensing (ACSS) and the Decentralized Cooperative Spectrum Sensing (DCSS) algorithms. Simulation results proved that the proposed algorithm overcome both algorithms in terms of spectrum the efficiency, the fairness and the throughput loss ratio. The proposed algorithm works whenever an empty slot is sensed, but the derived equations are too complicated to be applied in real time, especially with the increase of IoT devices number. Also the number of IoT devices used to evaluate the protocol is too small.

In [77], the authors addressed the problem of Device to Device (D2D) communication for IoTs devices considering priority, where the two devices need to meet each other on a common available channel. To achieve that, they proposed a priority channel hopping technique. The technique divides the time into slots, each serves a number of IoT devices and other devices. At the start of a time slot, the IoT device senses the network to obtain the available channel set, i.e. the channels not used by any other device at that time. The time to rendezvous is the time spent by an IoT device source to meet with its destination in a time slot. This time is to be minimized for a priority packet. In case of a non-priority packet, only the source do the search, while the destination only listens, the source puts the packet on a free channel, while the destination only scans the channels until it finds its packet on the right one. But in case of a priority packet, both the source and destination work together to find the free channel they should communicate on. As the rate of channels and IoT devices are different, the time to sense the channels are not equal, so an IoT device can sense only one channel in a slot, while another one can sense more than one. This leads to a problem of not recognizing the exact time of rendezvous. The authors then derived an equation to unify the hopping moment for all IoT devices. This mapping between the sensed channels for both the source and destination leads to the right channel they should communicate on.

Comparison between this technique and an old one named enhanced jump stay protocol was done. Mathematical and simulation proved that both techniques achieved 100% successful rendezvous rate, but the proposed protocol acquired less delay for high priority packets compared to the other one. The main disadvantages of this method are, first the processes performed by each IoT device are too

complicated so they need extra capabilities. Second, the number of used IoT devices to evaluate the protocol is small. Third, the energy consumption increases when the number of channels is more than fifteen.

The authors in [78] also addressed the problem of rendezvous between two SUs in CRIoTs network, but in an asymmetric asynchronous manner. The concept of Round Robin Tournament (all-play-all tournament) is used to guarantee rendezvous between the communication pair. It is a fair tournament, as each SU plays every other SU. It gives all the SUs the same opportunities and prevents one bad game from eliminating a SU from the competition. Simulation results proved that the proposed protocol performs better than the old existing channel hopping algorithms in terms of the average time to rendezvous, the degree of rendezvous and the throughput. The main disadvantages of this method are, first the process performed by each IoT device is extremely complicated so it needs special capabilities for the IoT devices. Second, the number of used IoT devices to evaluate the protocol is small.

In [79] a scheduling mechanism based on deep learning was proposed. The authors suppose that packets are stored in multiple buffers, and there exist multiple channels to be used. Their aim is to reach an adequate strategy to send packets from the buffers over the channels. A Markov decision process model was used, where a state represents the channel or buffer state, and an action represents the selected channel used to transmit the packets waiting their turn for the selected buffer. Authors supposed that channels and buffers take several states corresponding to the communication pair and transmission mode. In each time period and for regular states, packets are selected from a particular buffer to be sent over a particular channel. If the channel state is poor, packets are not transmitted. If a new packet arrives and the buffer is full, a chosen packet (may be the oldest) is lost. Deep learning (Q-learning) is used to map the transition between different states. It gets the optimal actions by continuous interaction with its surrounding area, using trial and error in a constant way.

For evaluation, the proposed scheduling is compared with the strategy iteration algorithm using simulation. Results proved that it has lower performance than

strategy iteration algorithm, but the complexity is extremely reduced. It also can work without any prior information. The main disadvantages of this method are, first the complicated processes to be performed at each IoT device, which is disliked due to IoT devices limited capabilities. Second, it has lower performance than the strategy iteration algorithm. Third the number of IoT devices in the simulation is too small. Finally, the buffer size should be carefully chosen to decrease the number of lost packets.

The Non-dominated Sorting Genetic Algorithm-II was used in [80], in order to solve the multi objective problem of maximizing both the spectrum utilization and the network throughput in a CRIoTs network. Considering multi-hop concurrent data flows, where a routing path will be composed of successive links. Graph theory was used to model the network considering nodes, links, spectrum channels and multi-hop concurrent data flows. Using this model, equations were derived in order to maximize both the throughput and the spectrum utilization considering signal-to-interference-plus-noise ratio. A solution of the problem is formed by successive spectrum channels (links) presenting the routing path. Different solutions (chromosomes) were chosen to form the initial population, and then the Non-dominated Sorting Genetic Algorithm-II was applied to reach the optimal solution. For evaluation, the proposed algorithm was compared with adapted Strength Pareto Evolutionary algorithm.

Simulation results proved that the proposed algorithm performs better than the other one. It reached better solution, with lower speed to reach this optimal solution using large and diverse population. The main disadvantages of this protocol are: first the number of IoT devices in the simulation is small. Second, genetic algorithms usually take a long time, which prevent the algorithm from working in real time, especially with high number of IoT devices, i.e. it is not scalable. Also the throughput rate does not increase as it should with the increase of the number of flows.

The objective of any spectrum allocation is to serve as many numbers of IoT devices as possible. A Guard Band (GB) is needed to avoid adjacent channel interference, when dealing with large number of low-cost IoT devices equipped with

CR-capabilities. Supposing the spectrum is divided into three types of blocks; idle blocks, data blocks and busy blocks (those used as guard bands). The technique proposed in [81] solved the problem of finding the optimal distribution of idle blocks among IoT devices. It attempted to maximize the throughput by reducing the number of assigned data-plus-GB channels, taking into account each device's demand rate, interference and total transmit power constraints. The authors proposed an algorithm to solve this problem using channel bonding and data aggregation, it consists of two phases, block assignment phase and release of extra channels post processing phase. For the block assignment phase, sequential fixing linear programming procedure is used. For the second phase, first the block with maximum size is selected. Then, the maximum number of channels starting from the left sided guard band of the block is released, taking into consideration the satisfaction of the rates conditions. Finally, the guard band is added again.

To evaluate the proposed protocol, it is compared with two older protocols. Simulation results proved that the proposed protocol achieved higher spectrum efficiency and also higher throughput than the two other older protocols. The main disadvantages of this protocol are, first as it is an NP-hard problem, so the solution is suboptimal. Second, the number of used IoT devices to evaluate the protocol is too small.

Table 4.1 summarizes the CR-IoTs protocols mentioned above. Note that there is a common disadvantage of the distributed ones that is the needed processes are too complex to be performed by the IoT devices. Also in all protocols, the number of IoT devices used for evaluation is small, sometimes even too small (about 10) which is not logic, as usually the number of IoT devices to be served in an area is in the order of thousands.

Table 4.1 Comparison of the CRIoTs Spectrum Sharing/Allocation Protocols

Protocol	Proposed idea	Sharing/ Allocation	Centralized/ Distributed	Spectral efficiency	Interference	Delay	Throughput	Packet Delivery Ratio	Energy Consumption	Fairness	Compared with each Other
B.Zhang et al. (2013) [70]	PU's lease their surplus portion of spectrum to IoT's using Nash Bargaining Solution	Sharing	Distributed					Improved using a multi-objective function			None
B.Moon (2017)[71]	Both PUs and IoT's work together to improve their performance, by deriving the blocking probability for the PUs and carried traffic for the IoT's.	Sharing	Centralized			Improved	Improved	Improved			None
H.Kim et al. (2017) [72]	Based on the conditional interference distribution, using stochastic geometry, derive the optimal transmission probabilities for all the IoT's.	Sharing	Distributed	Improved							Aloha
J.Wen et al. (2018) [73]	Find the optimal positions of the secondary information gathering stations used to collect the information for the IoT's, then find their optimal operating channels. PSO is used	Sharing	Distributed	Improved using a multi-objective function							RRA
L.Qu et al. (2012) [74]	A new spectrum allocation system model using Homo Egnalis Social Model	Allocation	Centralized	Improved			Improved				None
R.F.Shigueta et al. (2014) [75]	Based on traffic history, the IoT's allocate the channels, using probability theory.	Allocation	Distributed		Improved	Improved	Improved	Improved			Random ZAP TABU
S.Kim (2017)[76]	Using inspection game to distribute the free sensed slots between various IoT's.	Allocation	Centralized	Improved			Improved			Improved	DCSS DBSA
X.Liu et al. (2017)[77]	Derive an equation to unify the hopping moment for all CRIoTs, in order to minimize the rendezvous time between two IoT's	Allocation	Distributed			Improved			Improved		EIS
S. Mohapatra et al. (2018) [78]	Use of Round Robin Tournament to guarantee rendezvous between the communication pair between two IoT's, in an asymmetric asynchronous manner.	Allocation	Distributed			Improved	Improved				JS - EJS SARCH FRCH
J. Zhu et al. (2018) [79]	Use Markov decision process-based model to describe the state transformation of the system, and Deep learning to solve the problem of packets transmission from different buffers.	Allocation	Distributed				Improved	Improved	Improved		SI - WL - RS
R.Han (2018) [80]	Consider multi-hop concurrent data flows, where successive links forms a routing path. Use non-dominated Sorting Genetic Algorithm-II to maximize a multi-objective function.	Allocation	Centralized	Improved			Improved				Strength Pareto Evolutionary
H. B. Salameh et al.(2019)[81]	Using linear programming to find the optimal distribution of idle blocks (Guard bands) among CRIoTs transmissions.	Allocation	Distributed	Improved			Improved				

## 4.4 Conclusion

Internet of Things (IoT) presents a new life style by developing various applications. However, in order to avoid the bandwidth scarcity, IoT needs integration of the Cognitive Radio Networks (CRNs) capabilities. CRIoTs attracted some attention, but still the work done in this area is underdeveloped compared to that should be done.

This chapter surveys most of the protocols that addressed the problem of spectrum allocation/scheduling for CRIoTs. It has been noticed that most of the research done on this area uses small number of IoT devices, which is not applicable in most of the IoT applications. As a new idea in order to increase the number of the IoT devices, edge nodes/fog nodes can be added to handle considerable numbers of IoT device, so it will be to be responsible for several functions such as scheduling and channel allocation on CRNs. This idea will be used together with proposing new centralized/ distributed packets' scheduling technique for CRIoTs in this thesis.

The next chapter will propose new cognitive radio channel prediction techniques, that will estimate the primary channels status whether idle or busy, which will serve as an input to be used by the proposed channel allocation technique for CRIoTs in later chapters.

**Chapter 5**

**Cognitive Radio Networks**

**Channel State Estimation**

**Using Machine Learning**

**Techniques**



# **Chapter 5**

## **Cognitive Radio Networks Channel State Estimation Using Machine Learning Techniques**

In interweave Cognitive Radio Networks (CRNs), monitoring the spectrum to detect the unused portions (holes) is done by the spectrum sensing function, however it consumes both time and energy. So, some protocols use prediction not only to estimate the channel availability to be used by the cognitive users (Secondary Users SUs), but also to estimate the Primary Users (PUs) appearance to reduce the collision with it. One of these protocols uses Hidden Markov Model (HMM) but in a very simple way [82]. So, it does not perform well in several cases.

The aim of this thesis is to develop a spectrum sharing protocol for CRIoTs. However when using CRNs, the used channel status must be first known. This chapter's aim is to propose a way to predict the channel state, as the first step before developing the new spectrum sharing protocols illustrated in the next two chapters.

In this chapter, two new protocols are proposed for predicting the availability of the cognitive radio channel. Both of them uses HMM in an advanced way, where two models are produced in the training stage. In the prediction stage, the first protocol uses Bayes theorem together with these two models, while the second one uses Support Vector Machine (SVM) with the two models HMM parameters.

The chapter is divided into four sections. Section 1 reviews some protocols proposed for channel status prediction for CRNs. Section 2 proposes the two new protocols to solve the prediction of the cognitive radio channel availability problem, which are evaluated in section 3. Section 4 concludes the work done in this chapter.

## 5.1 Related Work

This section presents some researches that addressed the problem of CRNs' channel availability prediction. Many protocols have been proposed, this section will briefly cover some of these protocols [82]–[88].

Some papers use neural networks for channel prediction in CRNs. As an example in [83], a neural network based Multilayer Perceptron (MLP) model is used. It consists of an input layer, an output layer and a few hidden layers. The training patterns are obtained by ordering the entire binary series representing the channel state history into input vectors of a specified length. Forward pass is used for each input vector presented to the input layer of the MLP. The outputs of the neurons in each layer are calculated proceeding from the first hidden layer to the output layer. The output of the neuron in the output layer is treated as an estimate of the corresponding desired value. The difference between the desired value and its estimate is considered as an error. In order to minimize this error, back propagation is used to adjust the inputs' weight. This process is repeated until reaching a specified error value or for a number of iterations. A specific threshold for the output is used to determine if the channel state is idle or busy.

The problem in the previous method is that it may suffer from the local minima problem, which slows down the training. In [84], a genetic algorithm is used together with the MLP to optimize its structure, and to reduce the aggressive weight structural pattern to avoid trapping in local optimal solution.

In [85], Extreme Learning Machine (ELM) was used, which is a kind of Single Hidden Layer Feed Forward Neural Network (SLFN). It randomly chooses the input weights, and analytically determines the output weights of SLFNs. The ELM algorithm consists of three steps. First, building the model of ELM neural network. The  $n$  channel states of time slots are input to the input layer, the weights and threshold are initialized randomly. Second, training the ELM neural network spectrum prediction model, using the history channel data as the training data. It chooses sine as the activation function. Finally, the ELM neural network performs spectrum prediction. With the model trained by the previous step, the test data is used as the input of ELM, and the output of ELM is regarded as a predicted state.

Having channel states observations ( $X_1, X_2, X_3, \dots, X_{L-1}$ ), the proposed model in [86] assumes that each element of this sequence can be independently corrupted by noise, so that the state can be inverted by a probability  $p$ . These observations were treated as independent Bernoulli-distributed random variables. Classical statistical methodology was used, where the main idea is to access the likelihood of the observed sequence as a function of  $L$  (observation number where the channel's state change) and  $p$ . Then, it finds the values for these parameters that lead to its maximization. If the estimated  $p$  is less or equal to  $1/2$ , the prediction window is fully used by the SU, since it is more likely that this window is made by 0's. Otherwise, the prediction window is simply not used by the SU.

In [82], the channel state prediction in CRNs is done using Hidden Markov Model (HMM). It just takes the old states of the channel and uses Baum Welch algorithm to produce output. It calculates the probability of having the next value as 0 or 1 and chooses the one with the highest value.

For more details about spectrum prediction techniques, we invite the reader to consult surveys about spectrum prediction techniques for CRNs in [87], [88].

## 5.2 The Proposed Protocols

As mentioned in [82], the channel state prediction in CRNs is done using Hidden Markov Model (HMM). It just takes the old states of the channel, and uses Baum welch algorithm [89] to produce an output. It calculates the probability of having the next value as 0 or 1 and chooses the one with the highest value. This method has a main disadvantage, that the HMM model is trained using the whole day, which is impractical as the primary network channel's usage vary all over the day, which lower its chance to predict correctly. It also neglects to measure some serious parameters, such as the probability of PUs' collision (in case of wrong prediction) and the difference between true (right) predictions percentages given different inputs, which are very important factors to evaluate such protocol.

The protocols proposed in this chapter is the first part out of three parts of this research that mainly aims to develop a spectrum sharing protocol for Cognitive Radio Internet of Things (CRIoTs), but when using CRNs, the status of the used channels must be known first, so channel state prediction protocols for CRNs are proposed in

this chapter. The main idea of this chapter is to perform CRNs channel state prediction using two machine learning techniques, Hidden Markov Model (HMM) and Support Vector Machine (SVM). Note that while building the protocol proposed in [82], that uses Hidden Markov Model (HMM) in a classical way, it has been found that it leads to a very high probability of PUs' collision (which may reach 40% and above), and also high difference between true predictions percentage given different inputs.

Of course, spectrum must be sensed to obtain and store the real states of the channel at each slot (named observations) in the learning stage. Then, the prediction technique will use these real sensed states, to train itself to predict the state of the next time slot. Also spectrum sensed data helps the prediction techniques to give more accurate decision.

The outputs of the sensing function – i.e. the actual list of observation data named X (sequence of 0's and 1's)  $(x_1, x_2, x_3 \dots \dots x_t)$  – are used as training samples. This list of observed data will be converted into a matrix using a specific window, referred to as the sensing matrix.

In the proposed protocols, the sensing matrix is divided into two sets (according to the last column in the matrix). The first set -named the idle set- includes the observations with rows ending by an idle state, and the other set -named the busy set- includes the rows ending with busy state. So the system is trained that after the appearance of a specific sequence of channel's states, the next state will be mostly a specific one (either idle or busy). Figure 5.1 illustrates an example of how to use the sensed data to obtain the two sets; the idle set and the busy one.

This section is divided into two subsections proposing two protocols for channel prediction. The first subsection describes prediction using Bayes' theorem and the 2-model HMM produced by the two mentioned sets. While the second one performs the prediction using a hybrid HMM and SVM model, where the parameters produced from the 2-model HMM are used to generate the feature vector used in the training and the prediction of the SVM model.

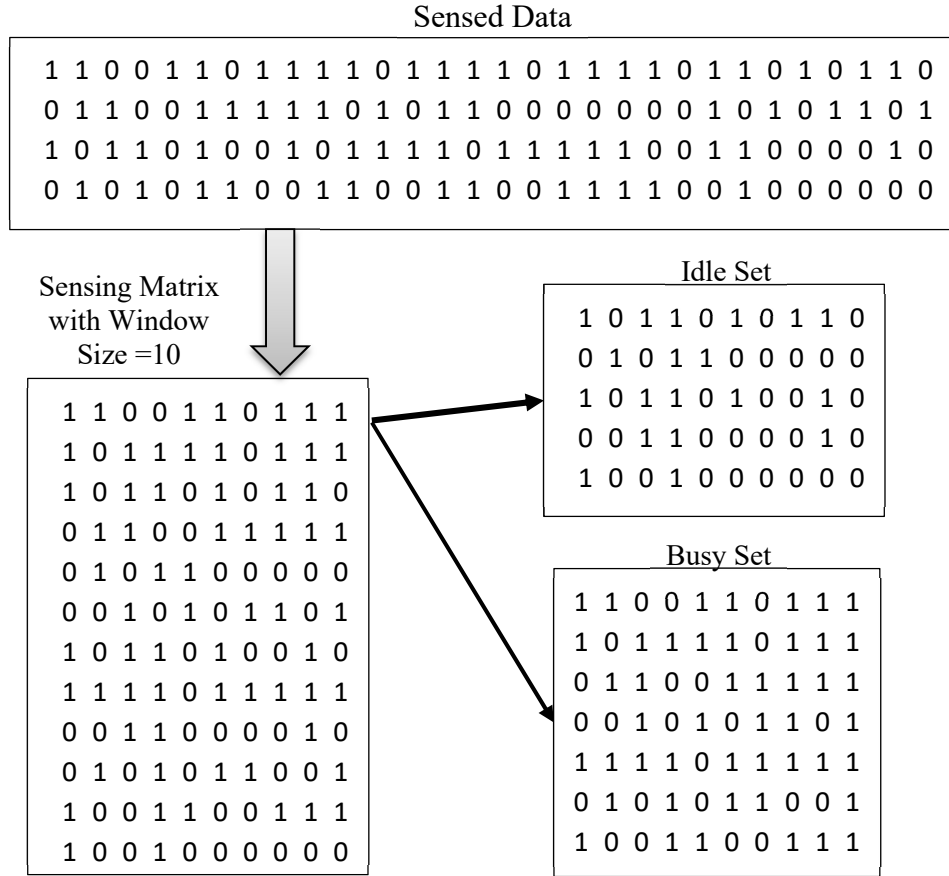


Figure 5.1 An Example of Obtaining the Idle Set and Busy Set from the Sensed Data

### 5.2.1 Prediction using Bayes' Theorem and HMM (2-Model HMM)

This protocol estimates the probability of the channel availability for the next period using Bayes theorem as follows:

$$P\left(x_t = \begin{matrix} idle \\ busy \end{matrix} \middle| x_1, x_2, x_3, \dots, x_{t-1}\right) = \frac{P(x_t)P(x_1, x_2, x_3, \dots, x_{t-1} | x_t)}{P(x_1, x_2, x_3, \dots, x_{t-1})} \tag{5.1}$$

Where:

$P(x_t)$  is the prior probability distribution.

$P(x_1, x_2, x_3, \dots, x_{t-1} | x_t)$  is the likelihood probability distribution.

$P(x_1, x_2, x_3, \dots, x_{t-1})$  is the joint probability distribution that gives the probability of each random variable in it. It acts as a normalization factor.

### A. Prior Probability Distribution $P(x_t)$

The prior probability distribution  $P(x_t)$  indicates the originally believed or the expert opinion about this quantity before any evidence is taken into consideration. It is calculated for both idle and busy state at time  $t$ , i.e. both  $P(x_t = \text{idle})$  and  $P(x_t = \text{busy})$  will be calculated.

As most of the days have approximately equal network load on a specific time, so, it is convenient to use the previous days' network load on that time to estimate the current channel state. It is calculated as the number of time slots in which the intended channel is either idle or busy for a window time in the previous days before the current day, divided by the total number of slots. This window considers the number of slots before and after the current one in the previous days.

$$P(x_t = \text{idle}) = \left( \frac{\text{Number of idle observations}}{\text{Number of total observations}} \right)_{\text{at window time}}$$
$$P(x_t = \text{busy}) = \left( \frac{\text{Number of busy observations}}{\text{Number of total observations}} \right)_{\text{at window time}}$$

or

$$P(x_t = \text{busy}) = 1 - P(x_t = \text{idle})_{\text{at window time}} \quad (5.2)$$

### B. Likelihood Probability Distribution

$$P(x_1, x_2, x \dots x_{t-1} | x_t)$$

The likelihood probability distribution can also be called as the conditional probability, which is the degree of belief in a specific evidence, given that the proposition is true.

The Hidden Markov Model [82] (HMM) is used to calculate the likelihood probability distribution, in order to predict the future state of a channel using probabilities based on the current and past state.

The first stage is the training stage, it uses the two sets produced from the sensing matrix explained previously to define two HMM models (idle model and busy model). The idle set will be used to train the idle HMM model, while the busy set will be used to train the busy HMM model.

HMM is a statistical Markov model in which the system being modeled is assumed to be a process with hidden (i.e. unobserved) states. Having N states and M hidden states, the HMM model's inputs are:

- List of observations  $O(O_1, O_2, O_3, O_4 \dots \dots O_t)$ , used as training samples.
- Initial HMM parameters  $\lambda (A, B, \Pi)$ :
  - A: transition probability NxN matrix.
  - B: emission probability NxM matrix.
  - $\Pi$ : Initial state Nx1 vector.

Training of HMM is done using Baum-welch algorithm [89], to adjust  $\lambda (A, B, \Pi)$  that maximize  $P(O|\lambda)$ , as follows:

$$P(x_1, x_2, x \dots \dots x_{t-1} | x_t = 0) \quad (5.3)$$

$x_t=0$  means the channel at time t is idle (predicted using the idle HMM model).

$$P(x_1, x_2, x \dots \dots x_{t-1} | x_t = 1) \quad (5.4)$$

$x_t=1$  means the channel at time t is busy (predicted using the busy HMM model).

Substituting in equation 5.1 using equations 5.2, 5.3 and 5.4 to apply Bayes' theorem:

$$P(x_t = idle | x_1, x_2, x_3 \dots \dots x_{t-1}) = \frac{P(x_t = idle) * P(x_1, x_2, x \dots \dots x_{t-1} | x_t = 0)}{P(x_1, x_2, x_3 \dots \dots x_{t-1})} \quad (5.5)$$

$$P(x_t = busy | x_1, x_2, x_3 \dots \dots x_{t-1}) = \frac{P(x_t = busy) * P(x_1, x_2, x \dots \dots x_{t-1} | x_t = 1)}{P(x_1, x_2, x_3 \dots \dots x_{t-1})} \quad (5.6)$$

The probability with the highest result will identify the next slot channel state.

Having the two HMM models  $\lambda_{idle}$  and  $\lambda_{busy}$ , each model calculates the likelihood probability distribution of this sequence ending with its corresponding state. The idle model will use the sequences ending with idle state, while the busy model uses the sequences ending with busy state.

Figure 5.2 represents the proposed prediction system. The first stage is the learning stage, it produces the two models  $\lambda_{idle}$  and  $\lambda_{busy}$  as explained above. The second stage is the prediction stage. It takes three inputs, the first two are the output of the first stage, i.e. the produced two models (the idle model and busy model). The third one is a short history, i.e. previous states of the channel before time  $t$ . At time  $t$  either the channel state will be idle or busy.

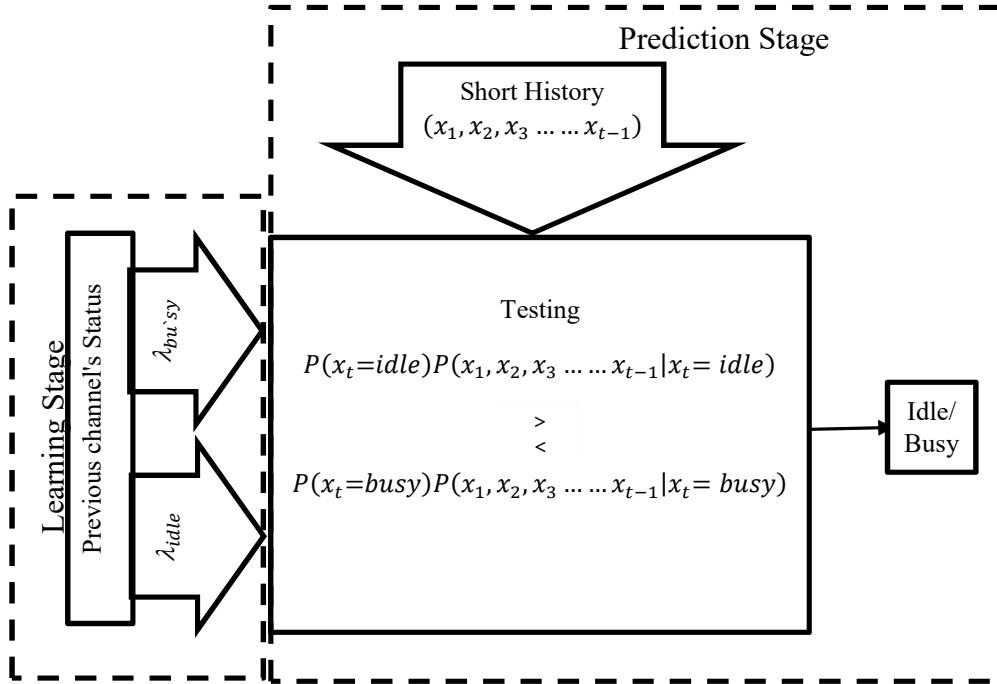


Figure 5.2 Prediction System using Bayes Theorem and HMM

### 5.2.2 Prediction using SVM with HMM parameters of the two models (2-Model HMM-SVM)

Support Vector machines (SVMs) [90], [91] are supervised machine learning algorithms used for classification and regression. Our protocol uses SVM for classification; it is done by analyzing the given labeled training data. SVM also works in two stages training and prediction. In the training stage, the given labeled feature vectors produced from training data are plotted in  $n$ -dimensional space, where  $n$  represents the number of features in the feature vectors. Then, the SVM finds an optimal separating hyperplane or a set of hyperplanes that maximizes the margin of the training data in order to differentiate well the presented classes. Figure 5.3 shows examples of linear and nonlinear hyper plane for SVM.



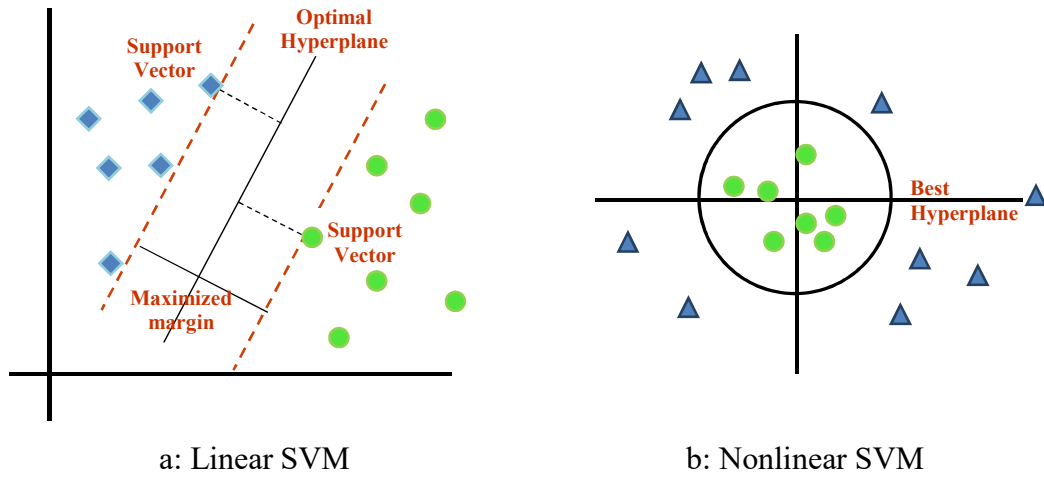


Figure 5.3 Support Vector Machine Hyperplane Examples [92]

In the prediction stage, the feature vectors produced from testing data are mapped to the same  $n$ -dimensional space. Then, based on its location the prediction is done to decide to which class the data belong.

The feature vector generation is done for both the training and the prediction stage. Before the training stage the observed data is divided with a window as in the previous model producing the sensing matrix, then a feature vector is generated for each row in this matrix.

The generation of the feature vector is done for each period of time of the observed sensing matrix, using the final transition and emission matrices for both the idle and busy models (resulted in the previous section). In other words, the feature vector will be divided into two parts: the first part includes the parameters calculated from the idle HMM, and the other part includes the parameters calculated from the busy HMM. Then, this feature vector will be referred to one of the two classes either the idle or the busy depending on the last state of this observation.

For each observation from slot  $t=1$  till slot  $t=T$  the feature vector parameters of the entire observation are the expected number of transitions from state  $i$  ( $\sum_{t=0}^T \gamma_t(i)$ ), and the expected number of transitions from state  $i$  to state  $j$  ( $\sum_{t=0}^T \xi_t(i, j)$ ). In other words,  $\gamma_t(i)$  sums the probability of being in each state  $i$  at each time slot, it is calculated in equation 5.7.  $\xi_t(i, j)$  sums the probability of being in each pair of states  $(i, j)$  at the corresponding pair of slots  $(t, t+1)$ , which is calculated using equation 5.8.

These calculations are done two times one for the idle HMM model and the other for the busy one trained previously. Then, both results are concatenated in one feature vector and labeled by its class depending on the last states. Finally these feature vectors and their classes are given to the SVM to be trained.

$\gamma_t(i)$  and  $\xi_t(i, j)$ , are calculated according to [89] as follows:

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{P(O_t|\lambda)} \quad (5.7)$$

$$\xi_t(i, j) = \frac{\alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)}{P(O_t|\lambda)} \quad (5.8)$$

where:

- $\alpha_t(i)$  is the probability of getting the observation  $(O_1, O_2, \dots, O_t)$  and being in state  $i$  at time  $t$ .

$$\alpha_{t+1}(j) = [\sum_{i=1}^N \alpha_t(i) a_{ij}] b_j(O_{t+1}), \quad \alpha_1(i) = \pi_i b_i(O_1) \quad (5.9)$$

- $\beta_t(i)$  is the probability of the ending partial sequence  $(O_{t+1}, O_{t+2}, \dots, O_T)$  given that it started with state  $i$  at time  $t$ .

$$\beta_t(i) = \sum_{j=1}^N a_{ij} b_j(O_{t+1}) \beta_{t+1}(j), \quad \beta_T(i) = 1 \quad (5.10)$$

- $P(O_t|\lambda)$  is the probability of getting the observation  $O_t$  given the model  $\lambda$ . It acts as a normalization factor.

These parameters are calculated using the Baum welch algorithm [89] used in the first method.

In the prediction stage, the feature vector is generated for a short period of time before time  $t$  using the same equations, then handed to the SVM classifier to classify whether the next time slot  $t$  will be idle or busy.

### 5.3 Protocols Evaluation

In order to evaluate our two proposed protocols (the 2-model-HMM and the 2-model-HMM-SVM), we implemented them together with the classical HMM protocol used in [82], and then compared between their performance under various conditions. The implementation and the compression are done using Matlab, which is a

proprietary multi-paradigm programming language and numerical computing environment developed by MathWorks [93].

This section is divided into four parts. The first one illustrates the value of the used simulation parameters. The second one defines the used performance parameters. The third one illustrates the effect of changing various input parameters on the performance of the three protocols. The last section analyses the results.

### 5.3.1 Simulation Parameters

The following simulation parameters are used:

- Number of states:  $N$ ; represents the number of states used.
- Number of hidden states:  $M$ ; represents the number of hidden states used.
- Period length:  $P$ ; represents the number of sensed slots in a day.
- Number of days;  $D$ ; represents the number of days used in learning.
- Matrix Window (in slots):  $T$ ; represents the length of a row by which the period length is divided to form the sensing matrix.
- Short History:  $SH$ ; represents the length of previous sensed data  $0 \rightarrow t-1$  used in order to predict the channel state  $t$ .
- Prior Days:  $PD$ ; represents the number of days used in calculating the prior probability.
- Prior Window:  $PW$ ; represents the number of slots used in calculating the prior probability.

Note that normal distribution is used for probability of packet presence. Table 5.1 represents the values used in the simulation.

K-folds cross validation is used in order to reduce the bias in choosing training and testing sets for classification [94]. In k-folds cross-validation, the original sample data is partitioned into  $k$  equal sized parts, then the following is repeated for a number of times:  $k-1$  parts are used as training data, and the remaining part is used for validation, of course different  $k-1$  parts are used every time

Table 5.1 Simulation Parameters' Value

Parameter	Used Value(s)
Number of states N	2 Zero represents Idle state One represents busy state
Number of hidden states M	Changed from 2 to 8
Period length P	4320 (24x60x3)
Number of days D	200
Matrix Window T	Changed from 30 to 180
Short History SH	9
Prior Days PD	20
Prior Window PW	30

### 5.3.2 Performance Parameters

In order to evaluate the proposed protocols, it is implemented along with the classical HMM using Matlab. Then, a comparison is done between their performances under various conditions. The following three performance parameters are used for evaluation of the three protocols:

#### A. True Percentage

The true percentage represents the percentage of right channel's state prediction to the total number of prediction. It is better to obtain high true percentage.

$$True\ Percentage = \frac{Number\ of\ right\ prediction}{Number\ of\ states\ to\ predict} * 100$$

(5.11)

#### B. Inaccuracy

Inaccuracy measures the impact of changing inputs on the obtained true percentage. Having lower inaccuracy value means that the difference between true percentages resulted at each time is small, which is better.

$$Inaccuracy = \frac{\sqrt{\sum_{\forall folds} (True\ percentage\ value - Mean)}}{Number\ of\ folds}$$

Where mean is the mean of the true percentage values produced in each iteration using cross validation.

(5.12)

### C. Probability of Primary Users' Collision

Probability of PUs collision is a very important metric when using CRNs. It represents the number of false negative predictions to the total number of prediction, i.e. when the channel state is predicted to be idle while it is actually busy by a PU. This will leads to collision for this PU traffic in case of assigning it to a SU, which is the main constraint when using CRNs. As clear, the PUs' collisions cannot be prevented completely but we are trying to keep it as low as possible. In other words, it is better to have small number of false negative predictions, in order to lower the PUs collision probability.

$$\begin{aligned} \text{Probability of PUs collision} &= \text{Probability of False negative prediction} \\ &= \frac{\text{Number of false negative prediction}}{\text{Number of states to predict}} \end{aligned}$$

(5.13)

### 5.3.3 Simulation Results

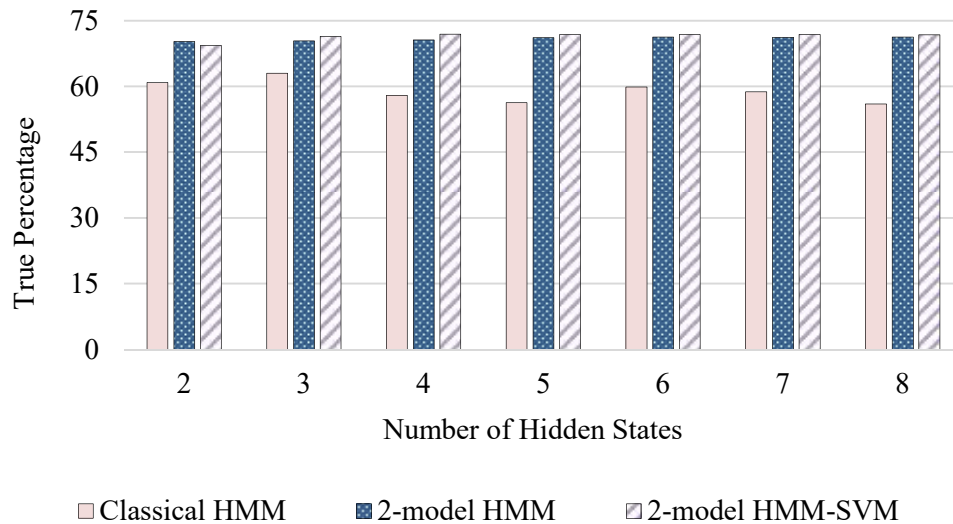
Some of the input parameters used in the simulator remains constant while others are changed in order to study their impact on the performance of the three protocols: the two proposed protocols (2-model-HMM and 2-model-HMM-SVM), and the classical HMM protocol in [82]. The three performance parameters; calculated using equations 5.11, 5.12 and 5.13 are measured while having different number of hidden states, different number of training days and different window size while applying the three protocols.

#### A. Changing Number of Hidden States

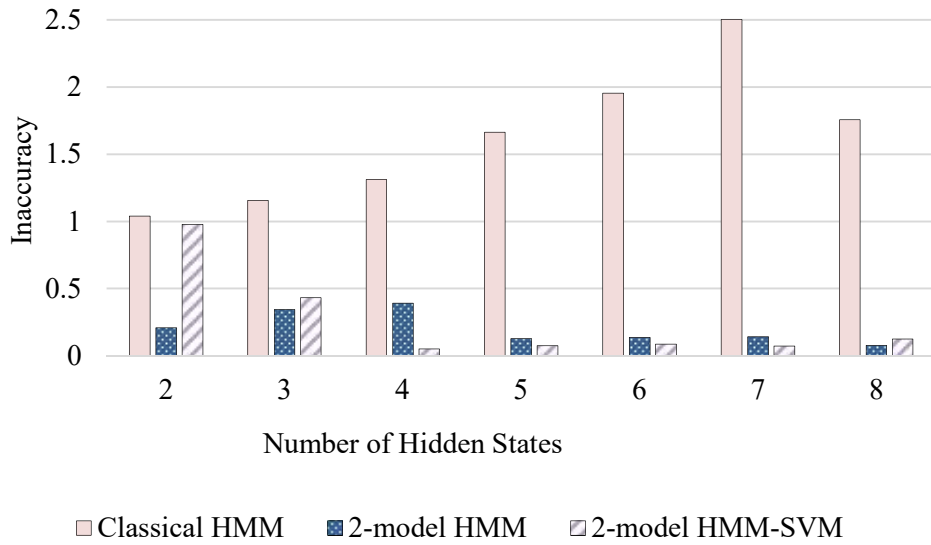
The charts in figure 5.4 plot the three performance parameters resulted from applying the classical HMM in [82], and our two proposed protocols (2-model-HMM and 2-model-HMM-SVM) while changing the number of hidden states. The HMM includes number of hidden states, which add more flexibility to the model, as it can be

used to find the features in the data that human cannot notice. In this case, the hidden states can be defined as different PUs habits, such as usage habits, different data rates, etc. From these charts, the following analysis and comments can be concluded:

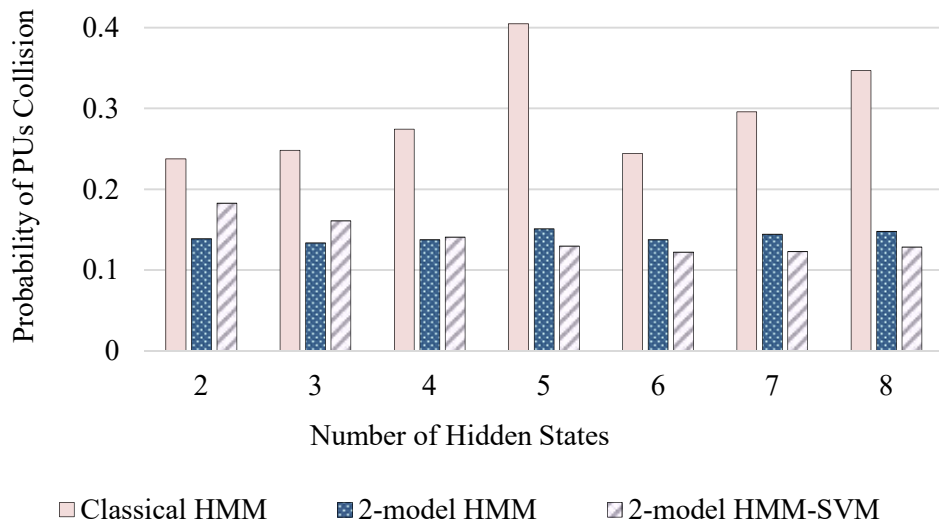
- The two proposed protocols perform much better than the classical HMM from all perspectives. This is logical as dividing the training data to produce sensing matrix into two sets resulting into two models, gives more flexibility to the training stage. Thus each model will be trained by the sensed slots ending by the model state. So, in the testing stage, giving a similar or near similar sequence, will lead to higher probability to predict the next slot channel's state.
- The number of hidden states has an impact on the proposed protocols' performance. As while using small number of hidden states the 2-model HMM performs better than the 2-model HMM-SVM, however it will not reach its best performance. While using large number of hidden states improves both protocols' performance, yet the 2-model HMM-SVM performs better than the 2-model HMM.



(a) True Percentage



(b) Inaccuracy



(c) Probability of PUs Collision

Figure 5.4 Performance Evaluation of the three Protocols Using Different Number of Hidden States

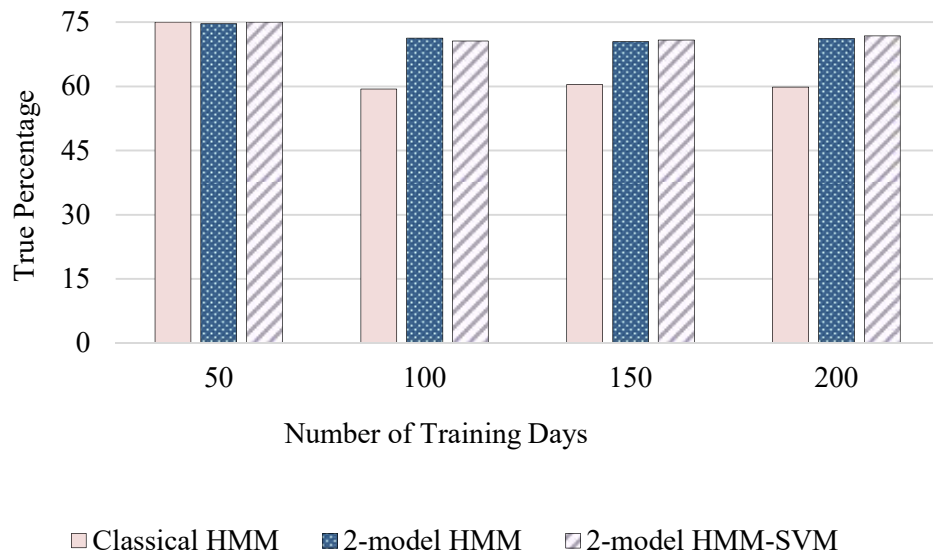
- Increasing the number of hidden states improves the protocols performance. This is logical as it discovers more hidden features in the data. However, increasing the number of hidden is done to a certain number till the model reaches a saturation level. The charts show that the performance of the two proposed

protocols is almost stable starting from using six hidden states. Note that this number cannot be calculated mathematically, it is found by training, i.e. by finding the number that on average gives the best results.

## B. Changing the Number of Training Days

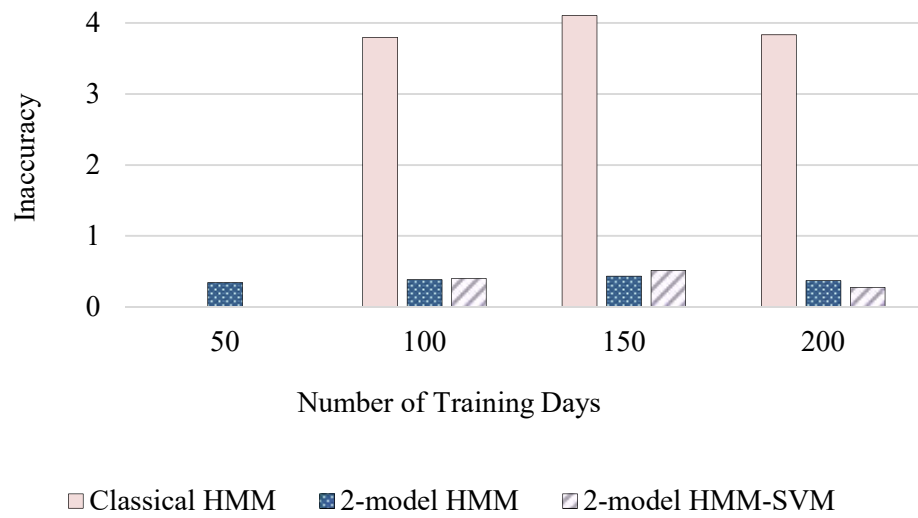
The charts in figure 5.5 plot the three performance parameters resulted from applying the three previously mentioned protocols, using different number of training days. From these charts, we can conclude the following points:

- Although the three protocols have the same high true percentage while using small number of training days (50), however they get very bad probability of PUs collision (about 25%) which is not acceptable. It is clear that the models are not trained well to recognize the PUs usage pattern.
- Increasing the number of training days, decreases the true percentage by a high rate while the probability of PUs collision decreases by a small rate for the classical HMM, the probability of PUs collision even increases again. On the contrary, increasing the number of training day decreases the true percentage by a small rate, and the probability of PUs collision by a high rate for the two proposed protocols.

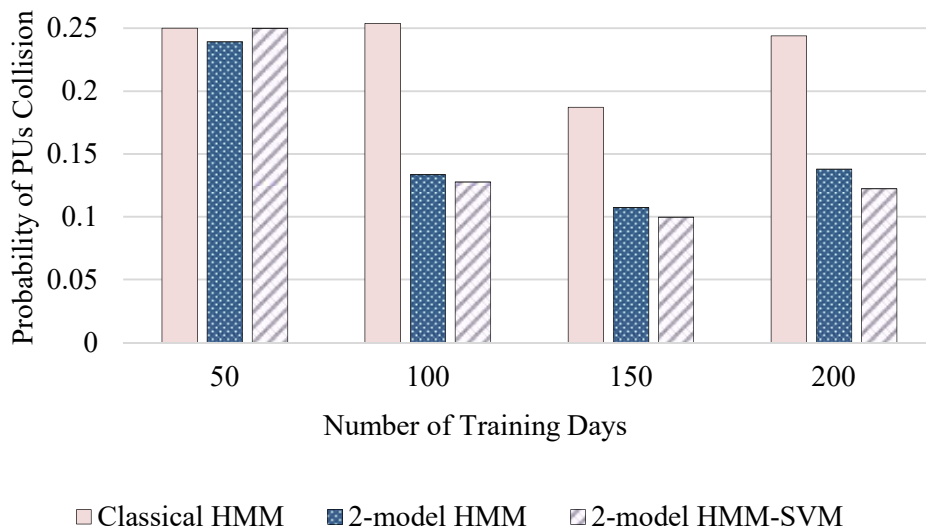


(a) True Percentage





(b) Inaccuracy



(c) Probability of PUs Collision

Figure 5.5 Performance Evaluation of the three Protocols Using Different Number of Training Days

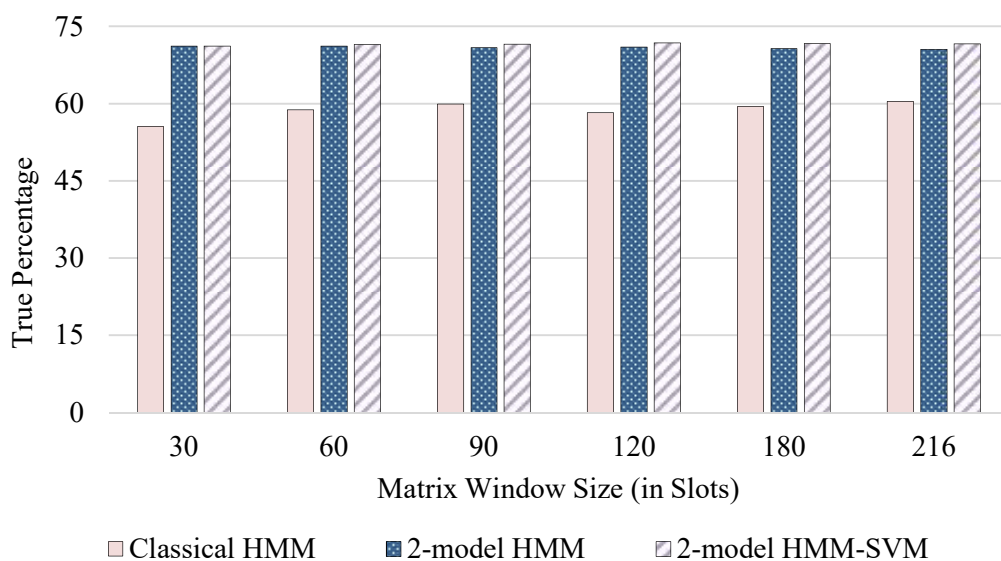
- Using high number of training days (from 100 to 200) has almost no impact on true percentage and inaccuracy using the two proposed protocols. But for the probability of PUs collision, the 2-model HMM-SVM performs better than the 2-model HMM. Using small (100) and high number (200) of training days leads to higher probability of PUs collision than using 150 training days. As for the 100

days the model is not trained enough, and for the 200 days the model is over trained but with a small percentage in both cases.

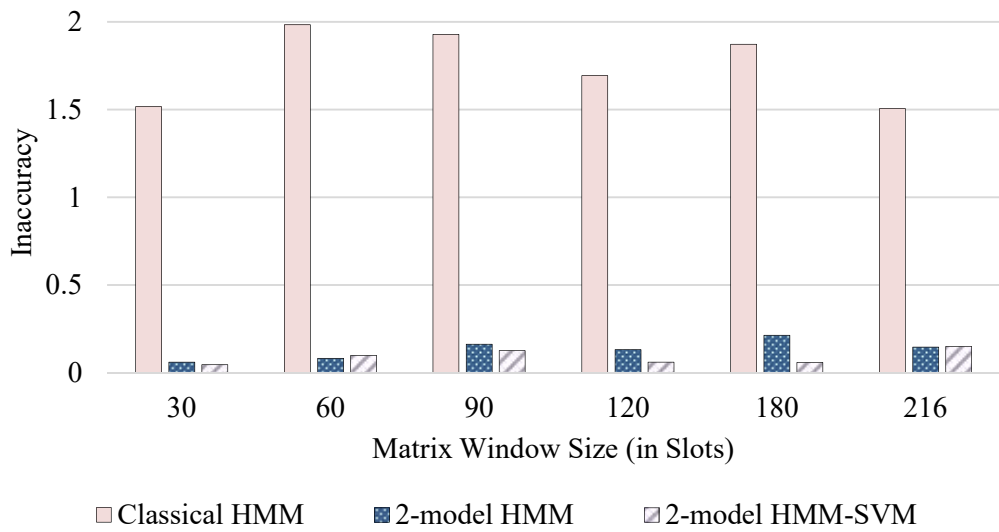
### C. Changing the Matrix's Window Size

The charts in figure 5.6 plot the three performance parameters resulted from applying the three protocols using different sensing matrix's window size. To obtain correct matrix with full row in the last row, values used are those where the period length are divisible by. From these charts, we can conclude the following points:

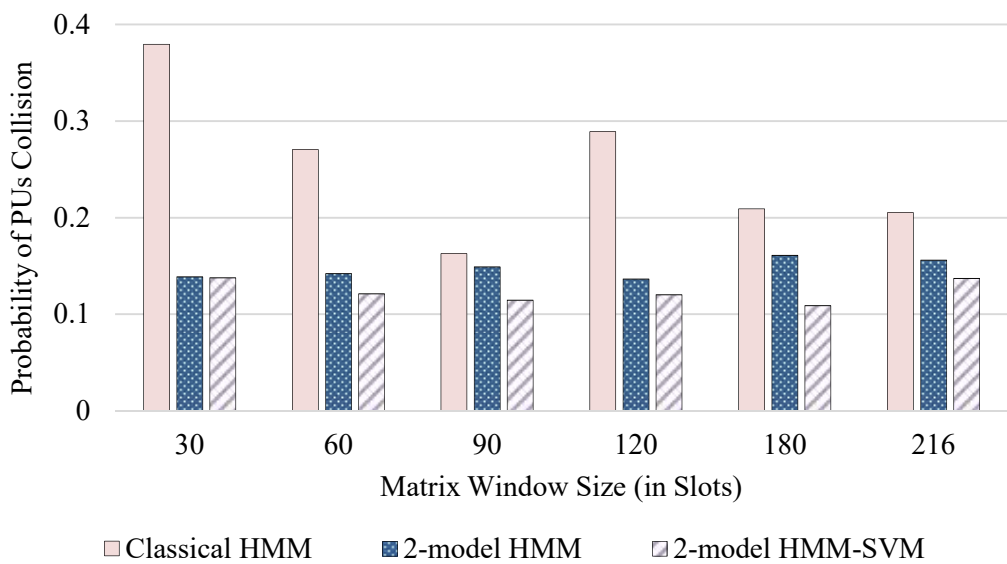
- The two proposed protocols perform much better than the classical HMM from all perspectives.
- Using different matrix window size has almost no impact on true percentage using the two proposed protocols. However for inaccuracy and probability of PUs collision, the 2-model HMM-SVM performs better than the 2-model HMM.
- For the 2-model HMM-SVM, using small matrix window size (30) and high window size (216) leads to higher probability of PUs collision than using average window size (60 to 180), while using small matrix window size leads to lower inaccuracy. As using a small or a large window size will not give an accurate PU usage pattern, so the model will not learn to anticipate this pattern.



(a) True Percentage



(b) Inaccuracy



(c) Probability of PUs Collision

Figure 5.6 Performance Evaluation of the three Protocols Using Different Matrix Window Sizes

### 5.3.4 Analysis and Drawbacks

Table 5.2 summarizes the results of the three protocols, it also illustrates the percentage of improvement when using the 2-model HMM and 2-model HMM-SVM protocols compared to the classical HMM protocol.

The two proposed protocols perform much better than the classical HMM from all perspectives. This is logical because dividing the training data to produce the sensing matrix, which is divided into two sets resulting two models, gives more flexibility to the training stage. Thus each model will be trained by the sensed slots ending by the model state. So, in the testing stage, giving a similar or near similar sequence will leads to higher probability to predict the next state.

In general, the 2-model HMM-SVM performs better than the 2-model HMM as it is trained with the HMM parameters, which are already trained. Also the feature vector produced for each sequence gives more precise information about the sensing slots in the sensing matrix.

Two main drawbacks exist for any channel prediction technique. The first is the necessity of waiting at least 100 days to obtain the sensed data (channel's states) the protocols will use in the training phase. The second is the amount of time and energy spent to train the model. The first drawback cannot be overcome as any prediction technique needs to have data to train the model. In our case the data takes time to collect, also cooperation between the primary and secondary network will be useful for this drawback. As for the second drawback although the training stage takes a long time, however, it will be done offline before the prediction stage. Moreover, the training will be done over a moderate to high computing machine. As this work is the first step in formulating a spectrum sharing protocol for CRIoT, the training and prediction will be done by a fog/edge node.

## **5.4 Conclusion**

In this chapter, we proposed two protocols to solve the problem of channel state prediction needed for interweave CRNs, the first one uses Bayes theorem and an obtained 2-model HMM, while the second uses SVM using the 2-model HMM parameters. For performance evaluation, simulations of both proposed protocols are done together with prediction using the classical HMM proposed in [82]. Three performance parameters were measured: true percentage, inaccuracy and probability of PUs collision (false negative prediction). For the three protocols, those parameters are measured while changing the number of hidden states, the matrix's window size and the number of training days. Results prove that the proposed protocols perform

Table 5.2 Percentage of Improvement using the Proposed 2-model HMM and 2-model HMM-SVM Protocols Compared to the Classical HMM Protocol

Window Size	Number of Training Days	Number of States	Changed Parameter	Performance Metric			Average Percentage of improvement of 2-model HMM over Classical HMM	Average Percentage of improvement of 2-model HMM-SVM over Classical HMM	
				Classical HMM	2-model HMM	2-model HMM-SVM			
	Average True Percentage	Average True Percentage	Average Probability of PUs Collision	58.961	70.843	71.396	20.15%	21.09%	
				1.626	0.204	0.259	87.48%	84.07%	
				0.293	0.142	0.141	51.64%	51.83%	
	Average True Percentage	Average True Percentage	Average True Percentage	Average Probability of PUs Collision	63.660	71.908	72.054	12.96%	13.19%
					2.932	0.384	0.298	86.91%	89.83%
					0.234	0.155	0.150	33.88%	35.84%
	Average True Percentage	Average True Percentage	Average True Percentage	Average Probability of PUs Collision	58.725	70.977	71.600	20.86%	21.92%
					1.750	0.134	0.091	92.36%	94.82%
					0.253	0.147	0.123	41.72%	51.18%

better than the classical HMM in several conditions. Also, the 2-model HMM-SVM outperforms both the classical HMM and 2-model HMM.

Because of the huge growth in IoT applications, unless it is integrated with CRNs, it will suffer from lack of spectrum and will not grow as expected. This chapter proposed the first step out of the three steps of this thesis. The training and prediction process proposed in this chapter are done by a fog node. The output of the prediction process will be used as an input to the next two steps, which will formulate a centralized and a distributed spectrum sharing technique for CRIoT, using machine learning techniques. The next chapter will propose a new centralized packets' scheduling protocol for CRIOts.

**Chapter 6**

**Centralized Packets**

**Scheduling Technique for**

**Cognitive Radio Internet of**

**Things Based on Discrete**

**Permutation Particle Swarm**

**Optimization**

# **Chapter 6**

## **Centralized Packets Scheduling Technique for Cognitive Radio Internet of Things Based on Discrete Permutation Particle Swarm Optimization**

The usage of Cognitive Radio Networks (CRNs) to establish Internet of Things (IoT) smart Networks is an effective and economical solution to the IoT spectrum scarcity problem. CRNs is merged with the IoT technology to be named Cognitive Radio Internet of Things (CRIoTs). Spectrum sharing/scheduling in CRIoTs has not attracted a lot of attention as it should do. This chapter proposes a new packets' scheduling strategy for CRIoTs. It is a centralized time slotted algorithm. It uses discrete permutation particle swarm optimization for scheduling packets among predicted primary network' empty time slots, resulted after applying the protocol proposed in the previous chapter. Health care facility can be taken as an example to apply the proposed protocol. An objective function is formulated to maximize the fairness index among fog nodes, minimize the packets' queuing delay and minimize the number of dropped packets at each time interval. In order to evaluate the proposed protocol, it is implemented along with another protocol named spectrum auction [95].

The chapter is divided into five sections. Section 1 reviews the Particle Swarm Optimization (PSO) technique and an old centralized scheduling technique. Section 2 formulates the scheduling problem. Section 3 proposes a new centralized packets' scheduling strategy for CRIoTs. Section 4 evaluates the proposed protocol by comparing its performance with that of another technique under various conditions. Section 5 concludes the chapter.



## 6.1 Background

This section consists of two main subsections. The first subsection reviews the Particle Swarm Optimization (PSO) computational method, which is used to iteratively optimize a problem. In the second one a review of an old centralized scheduling method named the spectrum auction technique is illustrated [95]. This technique is used for performance evaluation comparison.

### 6.1.1 Particle Swarm Optimization

Particle swarm optimization (PSO) is a population based stochastic optimization technique established by Dr. Eberhart and Dr. Kennedy in 1995 [96]–[98]. It imitates the motion of a flock of birds, in which flying birds arrange themselves in specific shapes or formations, and if a change in direction starts in the flock, it then “spreads through the flock in a wave.”

#### A. Standard Particle Swarm Optimization

PSO is a computational method that tries to find an optimal solution to a problem. It solves the problem by initially having a population of applicant solutions, called particles. Then, the location of each of these particles is changed to a new position (solution) based on a mathematical formula that uses the particle's position, its velocity, its best known position, and the global best position known for all the particles. The new position of a particle is calculated as follows:

$$X_i^{k+1} = X_i^k + V_i^{k+1}$$
$$V_i^{k+1} = \omega V_i^k + c_1 r_1 * (Pbest_i^k - X_i^k) + c_2 r_2 * (Gbest^k - X_i^k) \quad (6.1)$$

Where:  $V_i^k$  is the velocity of individual  $i$  at iteration  $k$ .

$\omega$  is the inertia weight parameter.

$c_1$  and  $c_2$  are acceleration coefficients.

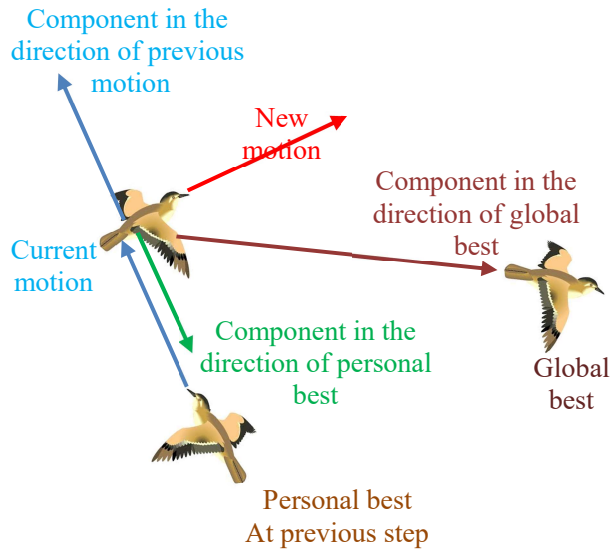
$r_1$  and  $r_2$  are random numbers between 0 and 1.

$X_i^k$  is the position of individual  $i$  at iteration  $k$ .

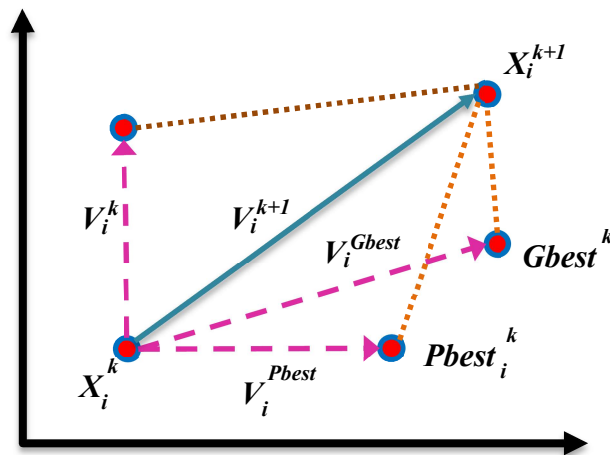
$Pbest_i^k$  is the best position of individual  $i$  till iteration  $k$ .

$Gbest^k$  is the best position of the whole group till iteration  $k$ .

Figure 6.1 shows the concept of modification of a searching point by PSO.



(a) Bird new motion



(b) New Position of the Particle

Figure 6.1 Concept of Modification of a Searching Point by PSO [98]

## B. Discrete Permutation Particle Swarm Optimization (DP-PSO)

As the particles that represent possible solutions in standard PSO give continuous values, so it is unable to handle discrete and binary problems. Some researches addressed this problem. For example the authors in [99], [100] proposed binary PSO, where the particles consist of only 0s and 1s. While in [100], [101] the authors proposed discrete PSO, where the particles consist of numbers chosen from a specific list of numbers, note that a number can be repeated, which is also not

suitable for our problem. Authors in [102] proposed a new version of discrete PSO algorithm that produces permutation particles, where each particle consists of unrepeated numbers chosen from a specific list of numbers.

Having a set of  $n$  packets ( $P_1, P_2, P_3, \dots, P_n$ ) needed to be sent, the problem addressed in this chapter is how to schedule these packets in a way that maximizes an objective function. So, this problem cannot be solved by regular PSO, binary PSO or normal discrete PSO. The suitable one is the discrete permutation PSO (DP-PSO) algorithm in [102] that produces a permutation sequence, i.e. results a list of non-repeated numbers at each iteration, which represent the non-repeated packets in our system. Discrete permutation PSO works as follows:

```

Generate an initial population (n particles)
  Evaluate each particle
    (calculate its objective function F(Pi))
  Repeat
  for each particle Pi
    do
      // Set its personal best
      If F(Pi) is better than F(personal best)
        Set the personal best to Pi
      // Set the global best position
      If F(Pi) is better than F(global best)
        Set the global best to Pi
      Update Pi Position using equation (11),
    end
  Until the stopping condition is met.

```

The position of the  $i^{\text{th}}$  particle  $P_i^t$  at iteration  $t$  is calculated as follows:

$$P_i^t = c_2 \oplus F_3(c_1 \oplus F_2(w \oplus F_1(P_i^{t-1}), P b_i^{t-1}), G b^{t-1}) \quad (6.2)$$

where:

$\lambda_i^t = w \oplus F_1(P_i^{t-1})$ , is the velocity of the particle.

$F_1$  refers to the insertion operator with probability of  $w$ .

$\delta_i^t = c_1 \oplus F_2(\lambda_i^t, P b_i^{t-1})$ , is the cognition part of the particle.

$F_2$  refers to the crossover operator with probability of  $c_1$ .

$Pb_i^{t-1}$  is the personal best of particle  $P_i$  till iteration  $t-1$ .

$P_i^t = c_2 \oplus F_3(\delta_i^t, Gb^{t-1})$  is the social part of the particle,  
it represents the collaboration among particles.

$F_3$  refers to the crossover operator with probability of  $c_2$ .

$Gb^{t-1}$  is the global best among all particles till iteration  $t-1$ .

## 6.1.2 Spectrum Auction Technique

Authors in [95] suggested using a spectrum auction to allocate spectrum to various Secondary users (SUs). The bidding problem is formulated as a Markov game. Each SU has a valuation function which may be calculated using any objective, such as an achievable capacity of a channel. Q-learning is used to decide any SU's bid at each time slot, using its state, and others SUs' states, estimated transition probability between states, etc. The auction runs as follows: At each time slot, first each SU calculates its value, and then decides its bid. After that; a central entity conducts the auction between various SUs among the available free slots. Finally various parameters are calculated for next round (slot).

Using the following notation and formulas:

- $SU_i$  uniformly decomposes the range  $[0, T]$  into  $N_i$  intervals as  $[T_0, T_1], [T_1, T_2], \dots, [T_{N_i-1}, T_{N_i}]$ , where  $T_0 \leq T_1 \leq \dots \leq T_{N_i} = T$
- The global state of the system at time slot  $t$  is defined as  $s^t = (s_i^t, s_{-i}^t)$
- $s_i^t$  represents the state of SU  $i$  at time slot  $t$ .
- $s_{-i}^t$  represents the state of other SU but  $i$  at time slot  $t$ .
- $b_i^t$  is the bid  $SU_i$  put at time slot  $t$ .
- $n \in [T_{n-1}, T_n]$ , it is used to set  $s_i^t$  and  $s_{-i}^t$  according to which win the bid.
- Transition probability  $q_{-i}$  is calculated as follows:

$$q_{-i}(s_{-i}^{t+1} = m | s_{-i}^t = n) = \frac{y_{n,m}}{\sum_m y_{n,m}} \quad (6.3)$$

- Q-function of  $SU_i$  at time slot  $t$  is calculated as follows:

$$Q_i^t(s_i, \tilde{s}_{-i}, b_i) = \begin{cases} (1 - \alpha_i^t) Q_i^{t-1}(s_i, \tilde{s}_{-i}, b_i) + \alpha_i^t (r_i^t + \delta V_i^t(s_i, \tilde{s}_{-i})) & \text{if } s_i^t = s_i, \tilde{s}_{-i}^t = \tilde{s}_{-i}, b_i^t = b_i \\ Q_i^{t-1}(s_i, \tilde{s}_{-i}, b_i) & \text{Otherwise} \end{cases} \quad (6.4)$$

where  $0 \leq \alpha_i^t \leq 1$  is the SU's learning rate

$r_i^t$  is immediate reward, it is the difference between the SU's valuation and its payment if the SU wins a channel.

The function  $V_i^t(s_i^t, s_{-i}^t)$  represents the value of the joint state  $(s_i^t, s_{-i}^t)$ , which is the expected discounted utility starting from that state. It is calculated as follows:

$$V_i^t(s_i^t, \tilde{s}_{-i}^t) = \sum_{s_i^{t+1}, \tilde{s}_{-i}^{t+1}} \left[ q_i(s_i^{t+1} | s_i^t, o_i^t) q_{-i}(\tilde{s}_{-i}^{t+1} | \tilde{s}_{-i}^t) \max_{b_i \leq m_i^{t+1}} \{Q_i^{t-1}(s_i^{t+1} | \tilde{s}_{-i}^{t+1}, b_i)\} \right] \quad (6.5)$$

The spectrum auction technique is formulated in the following steps:

- Initialize the  $Q_i$  values to zero for all possible states and bids.
- Initialize  $n(s)$  values to zero for all possible joint states  $s$ .
- For each time step  $t$  do
  - Observe the current state  $s_i^t$
  - With probability  $\epsilon(s_i^t, s_{-i}^{\sim t-1}) = c/n((s_i^t, s_{-i}^{\sim t-1}))$  choose a random bid, and with probability of  $1 - \epsilon(s_i^t, s_{-i}^{\sim t-1})$  use the following to place a bid.

$$\pi_i^*(s_i^t, \tilde{s}_{-i}^{t-1}) = \arg \max_{b_i \leq m_i^t} \left\{ \sum_{\tilde{s}_{-i}^t} q_{-i}(\tilde{s}_{-i}^t | \tilde{s}_{-i}^{t-1}) Q_i^{t-1}(s_i^t | \tilde{s}_{-i}^t, b_i) \right\} \quad (6.6)$$

- $n(s_i^t, s_{-i}^{\sim t-1})++$
- Observe the auction outcome  $o_i^t$  and receive the immediate reward  $r_i^t$
- Estimate the state of other SUs  $s_{-i}^{\sim t}$  and update the corresponding transition probabilities.
- Compute the value of state  $s_i^t, s_{-i}^{\sim t-1}$
- Update the  $Q$  value  $Q_i^t(s_i^t, s_{-i}^{\sim t}, b_i^t)$
- End for

## 6.2 Problem Formulation

The protocol proposed in this chapter is the second step in our research. It addresses the problem of scheduling of IoT devices' packets among the free slots, obtained by applying the cognitive radio networks' channel status estimation technique proposed in chapter 5. This problem is considered as an optimization problem. It is classified as a Non-Deterministic Polynomial-Time Hard (NP-hard) optimization problem [103], [104]. This section is divided into two subsections; the first subsection presents the mathematical formulation of the problem, while the second one explains the problem in details.

### 6.2.1 Mathematical Formulation

Few researches have addressed the problem of spectrum sharing and scheduling for CRIoTs. Chapter 4 has surveyed and reviewed most of these protocols according to our knowledge.

This chapter proposes a new centralized strategy to schedule the IoT devices' packets among the predicted free slots obtained by the new channel estimation technique proposed in chapter 5. Discrete-permutation particle swarm optimization is used. An objective fitness function that combines fairness, queuing delay and number of dropped packets is derived to be used to solve the problem as an optimization one.

Our system consists of IoT devices and fog nodes with different specifications. As mentioned in section 3.5, the fog nodes are used when designing and developing an adaptable and scalable IoT platform, as it introduces low latency and devices location awareness, real time decision making, network bandwidth utilization services.

Table 6.1 illustrates the variables' notation used while formulating the problem mathematically.

Table 6.1 Variables' notation

Parameters	Parameters Definition
$F_{NR}$	Number of fog nodes
$P_{ES}$	Predicted empty slot
$NP_{ES}$	Number of predicted empty slots
$C$	Number of channels
$T$	Time interval, consists of a number of successive slots.
$L$	Interval size in slots
$P_{i,j,RT}$	Packet generated by IoT $i$ , belonging to fog node $j$ at request time $R_T$
$o(P_{i,j,RT})$	Order of $P_{i,j,RT}$ in the waiting queue.
$P_{max_i}$	Number of slots after which the packet will be dropped if not scheduled.
$F_p(T)$	Fairness index per fog node at interval $T$ .
$a_j$	Number of allocated slots for node $j$ in interval $T$ .
$T.$	Previous intervals prior to $T$ .
$d_j$	Number of demanded slots for node $j$ from previous intervals that weren't scheduled.
$Q_p(T)$	Queuing delay in interval $T$ .
$N_p$	Number of packets sent in interval $T$ .
$AT_i$	Allocation time of packet $i$ .
$RT_i$	Request time of packet $i$ .
$Prior_i$	Priority of packet $i$ .
$D_p(T)$	Number of dropped packets in interval $T$ .
$Pr$	Priority level
$NS_{Prior k}$	Number of packets in the interval $T$ for priority $k$ that weren't scheduled.

## A. Objectives

$$\begin{cases} \text{Maximize } (F_p(T)) \\ \text{Minimize } (Q_p(T)) \\ \text{Minimize } (D_p(T)) \end{cases} \quad (6.7)$$

$$\text{where } F_p(T) = \frac{(\sum_j X_j)^2}{F_{NR(T-)} \sum_j X_j^2} \quad , \quad X_j = \frac{a_j(T)}{d_j(T-)} \quad (6.8)$$

$$Q_p(T) = \sum_{i=1 \rightarrow N_p(T)} (AT_i - RT_i) * Prior_i \quad (6.9)$$

$$D_p(T) = \sum_{Pr=1 \rightarrow \text{Highe priority}} \sum_{NS_{Pr}} Pr * NS_{Pr}(T) \quad (6.10)$$

## B. Decision Variables

$$\text{Pairs } (P_{i,j,RT}, P_{ES})_T$$

## C. Constrains

$$P_{max_i} < l * Prior_i \quad (6.11)$$

$$NP_{ES} \leq C * l \quad (6.12)$$

$$P_{ES} = \text{null if } \forall j d_j = 0 \quad (6.13)$$

$$\exists! P_{i,j,RT} \in (P_{i,j,RT}, P_{ES}) \ \& \ \exists! P_{ES} \in (P_{i,j,RT}, P_{ES}) \quad (6.14)$$

$$\text{If } Prior_i = \text{highest, } o(P_{i,j,RT}) = l \quad (6.15)$$

## 6.2.2 Problem Explanation

This subsection explains in details the problem formulated in the previous one. In our system there exists a number of fog nodes  $F_{NR}$ , each with a connected number of IoT devices. The predicted number of empty slots  $NP_{ES}$  is obtained using any CRNs channel status estimation technique. Our system uses the predicted empty slots obtained by applying the 2-model HMM-SVM protocol proposed in chapter 5 in the allocation process. Each empty slot is divided into successive intervals  $T$  with  $l$  time slots. At each time interval  $T$ , the fog nodes will have several packets  $P_{i,j,RT}$  received from their IoT sensor devices, which senses the patient's intended vital indicator such as blood pressure, temperature, number of heart beats, blood glucose, etc. Each fog node will assign every packet it received a priority ( $Prior_i$ ) according to the abnormality of the received vital sign. Note that the normal value of each vital signs



differs from one patient to another according to the patient medical case. The priority of each packet  $i$  will determine the maximum time  $P_{max,i}$ , it can be queued before being dropped if it wasn't allocated a channel.

Dealing with this problem as an optimization one, a fitness function was introduced. Three metrics were included in this function; the fairness index among the fog nodes requesting channels, the packets' queuing delay and the number of dropped packets (those exceeding their maximum time waiting in the fog nodes buffers without being scheduled). The goal of the fitness function is to maximize the fairness index, minimize the queuing delay and minimize the number of dropped packets.

Following are the explanation of objectives, constrains and decision variables defined in the previous subsection.

#### **A. Objectives**

We are dealing with a multi-objective optimization problem, with three objectives to optimize at each time interval (T):

- ***Maximize Fairness  $F_p$  (equation 6.8)***

The fairness index is calculated among the fog nodes requesting channels [105]. It is calculated for each interval based on the number of both demanded and allocated slots for each fog node. Its value ranges from 0 to 1.

- ***Minimize the queuing delay  $Q_p$  (equation 6.9)***

The queuing delay of each packet is calculated as the difference between the time when the slot will be allocated to the packet (in this interval) and the time it asked for it. It is then multiplied by the packet priority to emphasize its expected queuing delay when the packet priority is high.

- ***Minimize the number of dropped packets  $D_p$  (equation 6.10):***

Dropped packets are the packets that exceed their maximum time ( $P_{max}$ ) in the queue, so they will be dropped if they are not sent during this time interval. In order to decrease the probability of dropping high priority packets, each number of packets belonging to a specific priority is multiplied by its priority.

Solving the multi objective function, there are several methods, such as weighted sum method,  $\epsilon$ -constrains method, chebychev scalarization method, boundary intersection method, etc. [106], [107]. Weighted sum method was used for simplicity and rapidity. All the metrics in equations 6.8, 6.9 and 6.10 used in the multi objective function are united into one scalar. The multi objective function in equation 6.7 is transferred into the following single objective optimization function:

$$\begin{aligned} \text{fitness}(T) &= w_1 * F_p(T) + w_2 * (-Q_p(T)) + w_3 * (-D_p(T)) \\ \text{where: } w_1 + w_2 + w_3 &= 1 \end{aligned} \quad (6.16)$$

Positive sign is assigned to fairness with the aim to maximize it, in the contrary to the other two metrics that needed to be minimized, so they are assigned a negative sign. The first metric takes values from 0 to 1, so it is kept as it is, while the two other metrics can have any value, so they are normalized (to get a value from 0 to 1) before substitution.

## B. Decision Variables

At each time interval  $T$ , the allocation schedule produced a number of pairs  $(P_{i,j,RT}, P_{ES})$ , where:

- Number of produced pairs  $\leq NP_{ES}$ .
  - Inequality occurs if and only if the number of waiting packets  $< NP_{ES}$ .
- $P_{i,j,RT}$  is the packet generated.
  - $R_T < IT$ ; means that a request done at an interval, cannot be scheduled before the next interval.
- $P_{ES}$  is the predicted empty slot allocated to  $P_{i,j,RT}$ .

## C. Constrains

Constrain (6.11) determines the maximum time a packet  $i$  remains in the network to be allocated a slot, or it will be dropped. Note that as the dropping time of each packet is calculated by multiplying its priority with the number of slots in the time interval, so the packets with high priority have the right to stay longer in the network before being dropped. Moreover emergency packets cannot ever be dropped. Constrain (6.12) determines the number of predicted empty slots  $NP_{ES}$  at an

interval. Equality occurs if and only if all slots in the next interval are predicted to be empty. Constrain (6.13) states that an empty slot  $P_{ES}$  is left empty if and only if the packets' queue is empty. At constrain (6.14), for each pair  $(P_{i,j,RT}, P_{ES})$ ,  $P_{i,j,RT}$  can appear only once, means, which means that a packet is allocated only one slot, and  $P_{ES}$  can appear only once, i.e. an empty slot is used only once. Finally constrain (6.15) ensures that emergency packets (packets with the highest priority) are always added at the start of the queue, and can never be dropped. An emergency packet represents an abnormal vital sign, which must reach the central fog node and can never be dropped. As after receiving this emergency packet; the central fog node analyses it with the other packets, and may need a medical intervention which is represented in our system as action packets. These action packets are also marked as emergency one.

### 6.3 Proposed Protocol

The main idea of the proposed protocol is to make a centralized fair scheduling technique, for the huge amount of data received from IoT devices using Cognitive Radio Networks. Our proposed protocol is applied to a smart healthcare facility. The main issue in this system is the dynamicity of both the demands for resources and the predicted available channels at each time slot. In other words there exist different number of channels predicted to be available, they will be used to schedule different number of packets at each time slot. Another issue is the collision that may occur as a result of dealing with Cognitive radio Networks.

The problem is formulated in section 6.2 as a multi-objective optimization problem, then transformed into a single objective one that combines all objectives metrics presented in equation 6.16. It is solved using Discrete Permutation Particle Swarm Optimization (DP-PSO).

The architecture in Figure 6.2 configures the used hierarchical architecture of a smart healthcare system. This architecture is suitable for the smart healthcare facility (mentioned in section 3.4), as in regality the IoT sensors and actuators devices are connected to the patients, then those IoT devices will be connected to fog nodes, and finally a central fog node for the entire facility will be connected to all the fog nodes. Our architecture consists of three main blocks:

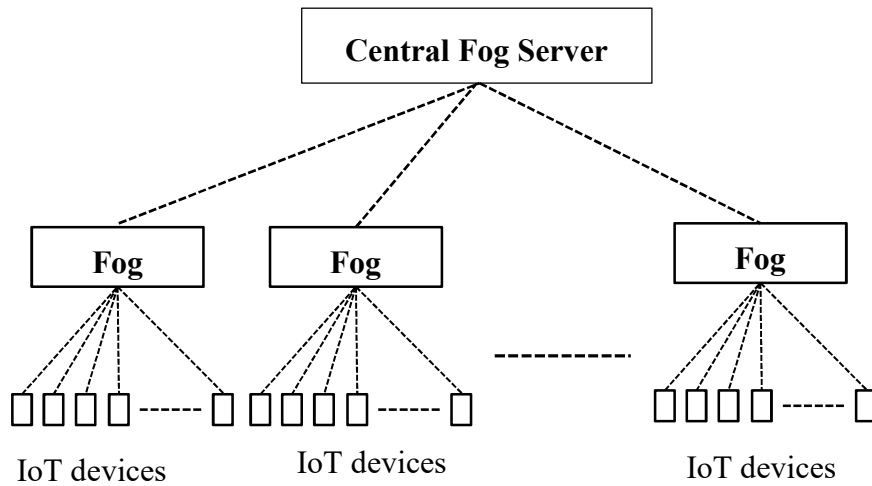


Figure 6.2 System Architecture

- IoT devices block:** This block consists of sensors and actuators. Each group of them is connected to a fog node, where they send data and receive orders. Note that each patient has a number of sensors and actuators (IoT devices) connected to him/herself. Each sensor senses one of the patient's intended vital indicator such as blood pressure, temperature, number of heart beats, blood glucose, etc., and sends it to the fog nodes connected to it. As for actuators they perform their intended task when they receive an order from the fog node. This order is formed by the central fog server depending on the sensed parameters sent by the sensors.
- First layer fog nodes block:** Fog nodes have many jobs. First, after receiving data packets from their connected IoT devices, they analyze and prioritize them. Our packet's system consists of five priority levels' packets going from 1 to 5 where 5 is the highest. Each data packet will be assigned a priority from 1 to 5 according to the normality of the sent vital sign. A normal value has a priority 1. The data priority increases with the abnormality of the vital indicator until reaching maximum priority of value equal to 5. The maximum priority represents very critical value (emergency) that needs an immediate medical intervention. Also, they form acknowledgments for emergency packets, i.e. those of priority equal to 5. It is known that the IoT device networks deals mostly with UDP packets that do not have any acknowledgements. However, here the packets are sent on a medium that may experience collision. So acknowledgments are formed for high priority packets only. Each fog node sends their requests to the central

fog server that forms the schedule. Then, after receiving the schedule from the central fog server, the fog nodes send the data packets according to that schedule. Furthermore, each fog node must drop the packets that exceeded their time in the queue and were not sent. Finally, they forward the needed actions to the IoT devices in case of received emergency packets.

- **Central fog server:** After receiving the requests from different fog nodes, the central fog server forms the schedule. It first adds the emergency and acknowledgement packets from the last interval, then applies the DP-PSO to solve the objective optimization function for the ordinary packets intended to be scheduled. After receiving the packets sent by the fog nodes depending on the schedule, the central fog server analyzes the data packets. Then, it forms its own packets intended for the actuators, and acknowledgements for the received emergency packets.

Our aim is to schedule the packets' requests received by the central fog server from the fog nodes, to be analyzed, processed and stored, together with the formation of acknowledgements for emergency packets and action packets.

The proposed protocol, named Scheduling based-on DP-PSO (SDP-PSO), is a time slotted protocol that applies DP-PSO in order to schedule packets for future slots (time interval), i.e. the scheduler would schedule the data packets requested in the past time interval (which consists of a number of slots). So, the optimization will be done for each interval instead of each time slot, which allows saving both time and energy. The time interval must not be too short to worth the time and the energy the optimization process will consume to find a near best solution. It also must not be too long so the overall delay will not be affected.

As explained in section 6.2, the proposed objective fitness function consists of three objectives, maximizing the fairness index among fog nodes, minimizing both the packets' queuing delay and the number of dropped packets (that exceeded their time in the fog node queue). Those parameters are considered significant when dealing with a channel scheduling technique.

Each particle in SDP-PSO that schedules a number of packets is treated in a similar way to the interval used in the problem formulation in section 6.2. At each iteration a number of particles are produced, each includes a number of pairs

consisting of the packet and the predicted empty slot allocated to it. For each particle in the generation, the three parameters; the fairness index, the queuing delay and the number of dropped packets, mentioned in the objective function equation (6.16); are measured, in order to conclude a near best allocation pattern derived after several iterations.

Figures 6.3 and 6.4 illustrate a block diagram and a sequence diagram for the proposed procedure SDP-PSO to perform the scheduling process:

1. Each IoT device senses and sends a packet to its connected fog node every time interval  $\Delta t$ , where  $t_{min} < \Delta t < t_{max}$ .
2. Each fog node collects, analyzes and prioritizes the packets received from various IoT devices, then queues them. Based on the packets priority, they will be divided into two types, ordinary and emergency packets.
3. Each fog node sends its channels' demands to the central fog server.
4. The central fog server collects the packets requests received from various fog nodes.
5. At each interval time, the central fog server forms the schedule based on the estimated free slots obtained from applying the protocol proposed in chapter 5. The schedule is formed as follows:
  - First add the emergency packets coming during the previous interval.
  - Then, add the acknowledgement packets formed during the previous interval. Note that they are formed only for sent emergency packets.
  - Finally, add the ordinary packets requests received from various fog nodes during the previous intervals. These packets are ordered by applying the SDP-PSO (explained in section 6.2) to schedule them using the fitness function in equation 6.16.
6. The central fog server sends the schedule to the fog nodes, along with its action emergency packets and acknowledgments from the previous interval time.
7. Each fog node sends all the packets (emergency and ordinary) according to the sent schedule.

8. Each fog node forms its own acknowledgements for the action packets sent by the central fog server, to be sent in the next interval time, i.e. added with the channels' request for next time interval.
9. Each fog node drops the packets exceeding their allowable time.
10. The central fog server collects and analyses the data in packets received from various fog nodes.
11. The central fog server forms its acknowledgements and action packets for the emergency packets, to be sent in the next interval time.
12. Each fog node forwards the actions packets sent by the central fog server to the IoT actuators (in case of emergency packets).

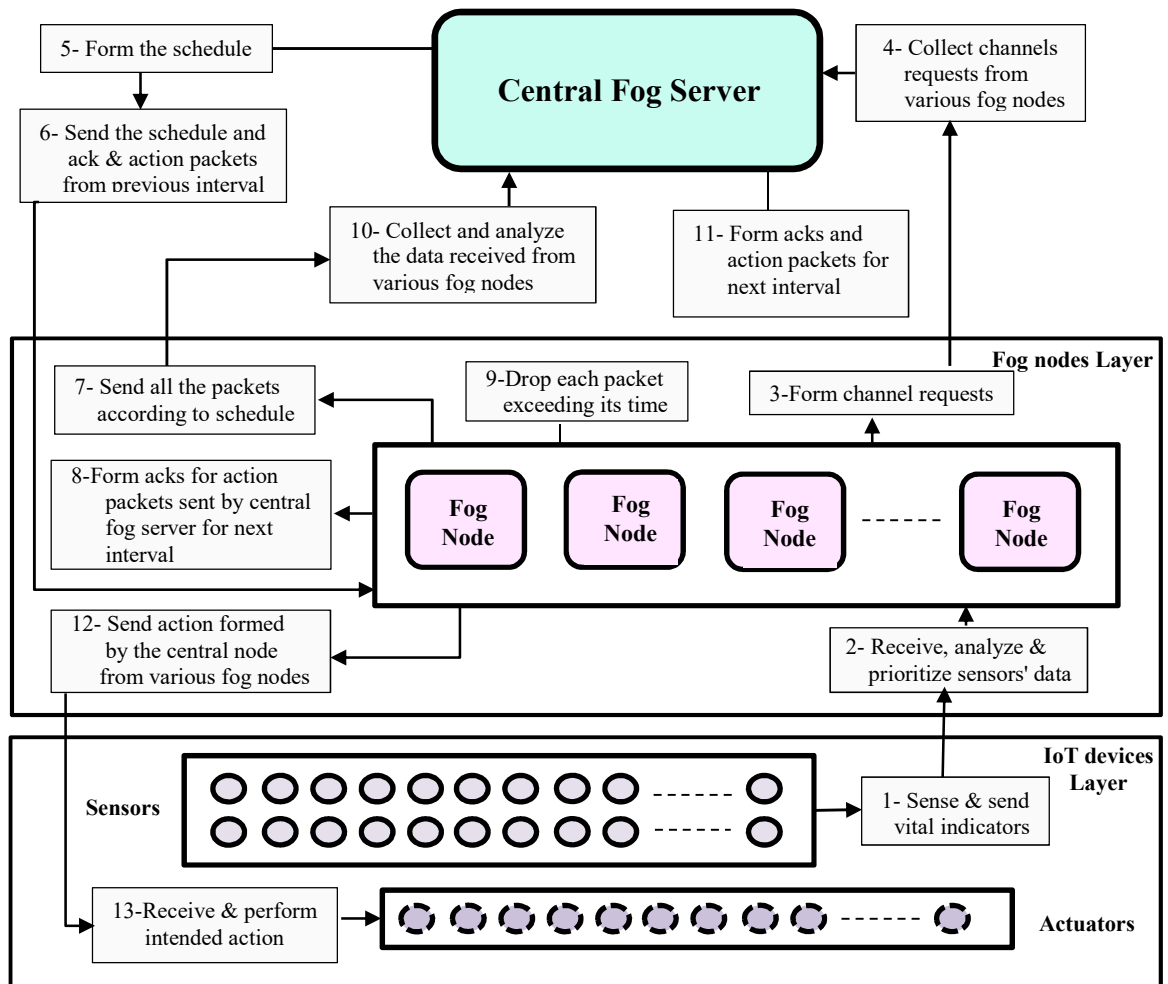


Figure 6.3 Block Diagram of the Proposed Protocol

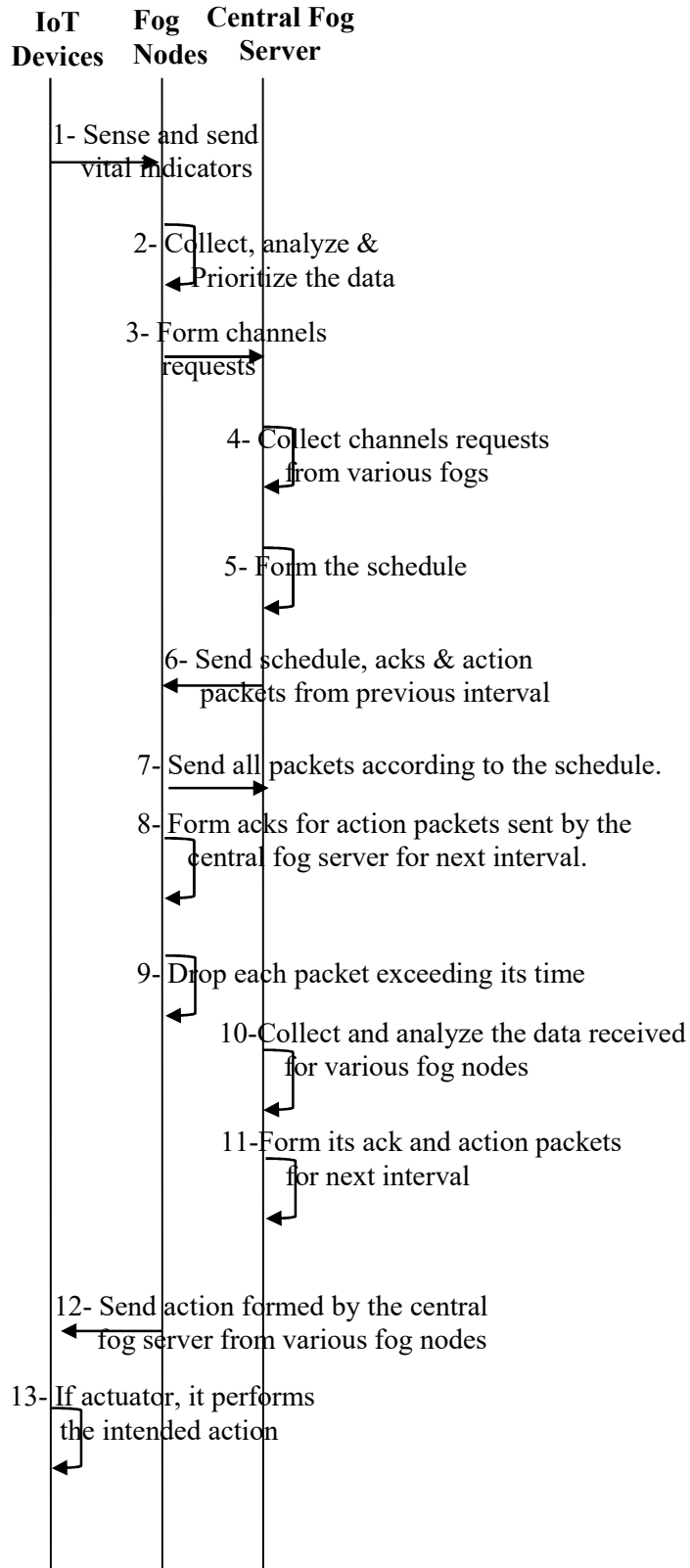


Figure 6.4 Sequence Diagram of the Proposed Protocol



13. Each actuator receives and performs the action it receives from the fog node.

Note that any packet that weren't scheduled or dropped in any time interval will be postponed to the next time interval.

Steps 1, 12 and 13 are out of our scope; our work is to perform the steps from 2 till 11.

## 6.4 Performance Evaluation

In this section, the proposed Scheduling based-on DP-PSO (SDP-PSO) protocol is compared with the spectrum auction technique in [95] explained in subsection 6.1.2. This protocol was used in the comparison as it is suitable to be customized and applied by using our objective function. This section consists of three subsections. The first part calculates the complexity of both protocols. The second one states the evaluation of the protocol using the implemented simulator. The third part analyses the results.

### 6.4.1 Protocols' complexity

This section estimates the time and space complexity of SDP-PSO protocol and the spectrum auction protocol.

#### A. Time complexity

The time complexity is defined as the computational complexity which describes the amount of time taken to execute an algorithm. It is usually predicted by counting the number of basic processes executed by the algorithm [108].

The time complexity of the spectrum auction protocol is  $O(S_i * N_i * B_i)$  [11], where  $S_i$  is the state space size for each node  $i$ ,  $N_i$  is the number of classes for other nodes' states and  $B_i$  is the bid space for node  $i$ . Note that increasing the value of these parameters results in more accurate results, but also increases the complexity.

While building the auction protocol, we used small values for each parameter,  $S_i=10$ ,  $N_i=10$  and  $B_i=20$ , so the auction has a time complexity of  $O(2000)$ .

In order to calculate the time complexity of SDP-PSO protocol, one of two main operations may dominate. The first one is the generation of the initial particles, we do not start with random particles, instead we use particles sorted according to some of the used performance metrics. So this operation has a time complexity of  $O(P*N*\log(N))$ , where  $P$  is the number of particles and  $N$  is the number of packets that needs to be scheduled at each time interval. The second operation is the crossover operation performed for each particle at each iteration. It includes search, elimination and insertion of a number of packets' index in a particle. This operation has a time complexity of  $O(I*P*N)$ , where  $I$  is the number of iterations. The values of  $I$ ,  $P$  and  $N$  determine which one of the two operations dominates.

While building the SPD-PSO protocol, we use small values for both the number of particles and the number of iterations, both are given a value of 5, but  $N$  is changed according to many parameters (number of IoT devices, number of channels, network load, etc.). Using the same value in table 6.2, we have an average value of  $N$  equals to 6 and maximum value equals to 42. Using both the average and the maximum value, the second operation dominates, with a time complexity of  $O(150)$  and  $O(1050)$  respectively.

So, the SDP-PSO protocol has a smaller time complexity than the spectrum auction protocol, it even runs every time interval (which consists of a number of slots), while the spectrum auction should be run every time slot which leads to impossibility of using it in real time.

## **B. Space complexity**

Space complexity is the amount of storage (memory) used by the algorithm to execute and produce the result.

The space complexity of the spectrum auction protocol is  $O(S_i*N_i*B_i)$ . It is the same as the time complexity, as each node needs to keep a table of size  $S_i*N_i*B_i$  [95]. Using the same values of  $S_i$ ,  $N_i$  and  $B_i$  parameters mentioned above, leading to space complexity of  $O(2000)$ .

In order to calculate the space complexity of SDP-PSO protocol, we need to check the space needed for the two main operations. The first one has a space complexity  $O(P*N)$ , while the second operation has a space complexity of  $O(N)$ .

Although the search is  $O(1)$ , we used another array in storage. Also, we need to store  $(P*N)$  particles. So, SDP-PSO protocol has a space complexity  $O(P*N)$ . Using the same values of the  $P$  and  $N$  parameters mentioned above, the space complexity is  $O(30)$  and  $O(210)$  for average and maximum value of  $P$ .

So the SDP-PSO protocol has a smaller space complexity than the spectrum auction protocol. Note that after each time interval, the SDP-PSO protocol does not need to keep anything except the packets to be sent in the next intervals. While the spectrum auction protocol needs to store the values of many parameters beside the packets to be sent at each slot, as most of the calculated metrics at each time slot depend on the parameters from the past time slot.

## 6.4.2 Simulation Evaluation

In order to evaluate the proposed SDP-PSO protocol, it is implemented along with the spectrum auction technique in [95] using Matlab. Then, a comparison is done between their performances under various conditions. This section consists of two parts. The first part presents the used evaluated metrics measured using the simulator. The second part shows the influence of varying different input parameters on the performance of the two protocols.

Note that, in order to help the optimization technique to find a near best allocation scheme faster, packets in each of the initial particles are ordered according to their request times, priorities, dropping times, etc. Also, for the initial personal and global best particles, we do not start the simulation with random particles, instead we use particles sorted according to some of the packets parameters that directly affect the evaluated metrics.

### A. Performance Evaluation Metrics

The three performance metrics used for evaluation are as follows:

#### a. *Fairness index*

This index must be maximized in order to allow the fog nodes to obtain fair shares of the estimated free slots.

$$F = \frac{(\sum_j X_j)^2}{F_{NR} \sum_j X_j^2} \quad (6.17)$$

$$\text{where } X_j = \frac{AS_j}{DS_j}$$

F is the fairness index among all the fog nodes.

As<sub>j</sub> is the total number of allocated slots for node j.

Ds<sub>j</sub> is the total number of demanded slots for node j.

F<sub>NR</sub> is the total number of fog nodes.

### ***b. Average Packet Queuing delay***

It represents the waiting time for the packets. The average queuing delay for all received packets must be as low as possible.

$$Q = \frac{\sum_{i=1 \rightarrow NR} (AT_i - RT_i)}{NR} \quad (6.18)$$

where Q is the average packet queuing delay

NR is the total number of received packets

AT<sub>i</sub> is the time the packet i is allocated a slot

RT<sub>i</sub> is the time the packet i requests for a lot.

### ***c. Percentage of Dropped Packets***

Ordinary packets are discarded (dropped) after remaining in the network for a specific time without being allocated a slot. This time is variable according to packet priority. In order to minimize the percentage of dropped packets, low number of dropped packets must be obtained.

$$D = \frac{ND}{N} * 100 \quad (6.19)$$

where D is the percentage of dropped packets.

ND is the total number of dropped packets.

N is the total number of packets.

## **B. Simulation Parameters**

The following simulation parameters are used:

- Number of slots: represents the number of slots used in the simulation. Note that not all slots are free all the time; instead some are occupied by primary users. The

free slots are predicted by the applying the protocol proposed in chapter five.

- Number of fog nodes  $F_{NR}$  + Central fog server: represents the number of fog nodes used in addition to the central fog node.
- Number of IoT devices per fog node: represents the number of IoT devices per fog node.
- Number of channels, C: represents the number of used channels.
- Interval by which IoT devices send packets: to indicate the packets' data rate. Small number indicates high data rate, data rate decreases with the increase of this number.
- Number of slots after which the packet will be dropped, Pmax: it is a variable number that represents the number of slots an unsend packet is allowed to stay in the network before being dropped. It is variable according to the priority of the packet, it is calculated by multiplying its priority with the number of slots in the time interval, so the packets with high priority have the right to stay longer in the network before being dropped.
- Interval size, l; represents the interval size in slots.

Table 6.2 represents the values of the parameters used in the simulation.

Table 6.2 Simulation Parameters' Value

Parameter	Used Value(s)
Number of Slots	72000
Number of fog nodes $F_{NR}$ + Central fog server	3
Number of IoT devices per fog node	Changed from 100 to 700
Number of channels, C	Changed from 3 to 5
Interval by which IoT devices send packets	Changed from 0.5 to 10 mins interval till 0.5 to 1 min interval
Number of slots after which the unsend packet will be dropped, Pmax	Packet priority * interval size
Interval size in slots, l	3

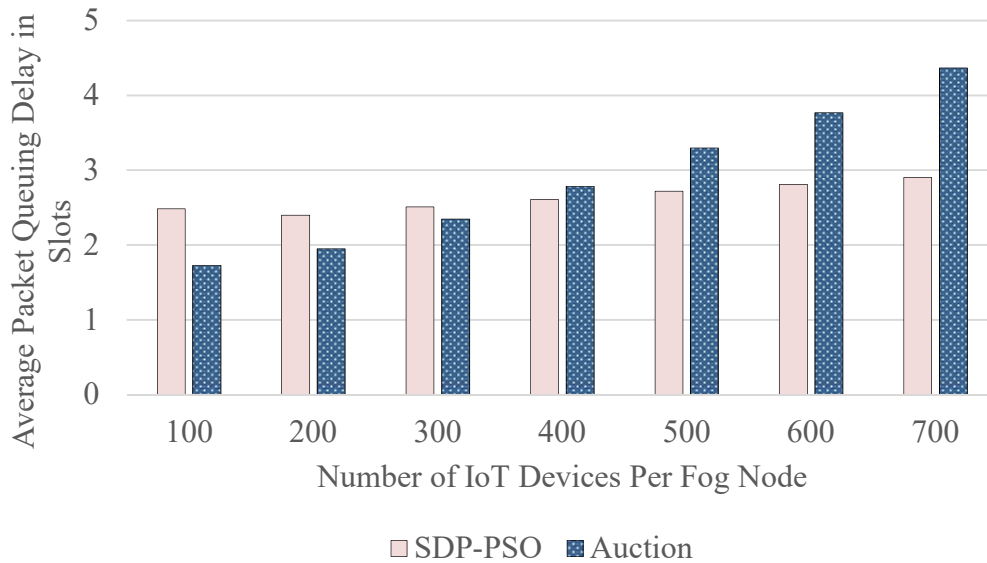
### C. Simulation results

The three performance evaluation metrics in equations 6.17, 6.18 and 6.19 are measured. For the simulations, some parameters remain constant, while others are variable in order to study the impact of their change on the performance of the two protocols: the proposed SDP-PSO and the spectrum auction in [95].

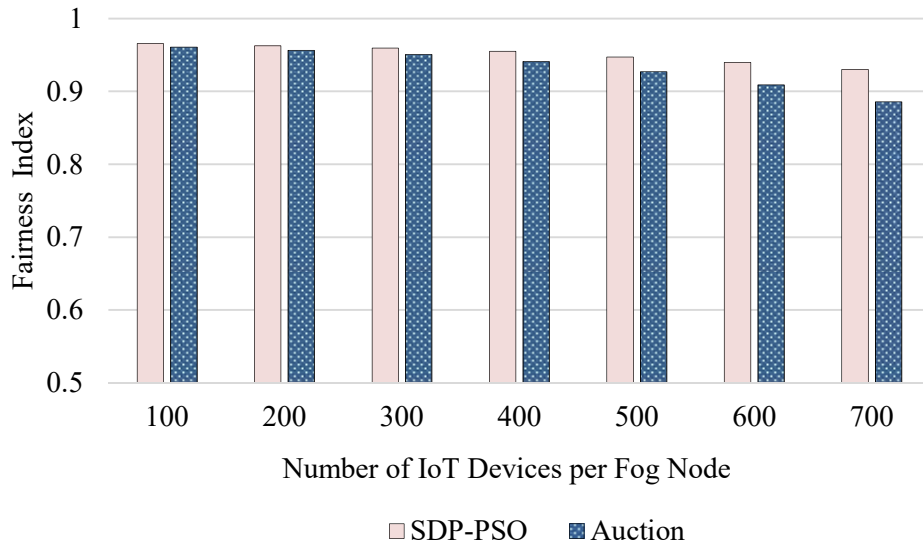
#### a. Changing number of IoT devices per fog node.

The graphs in Figure 6.5 illustrate the three evaluated metrics resulted when applying both protocols, using various number of IoT devices per fog node, varied for 100 to 700. The number of channels and the interval by which each IoT device sends packets are kept constant, with values 4 and 0.5 to 5 mins respectively. The following points are the conclusion driven from these graphs:

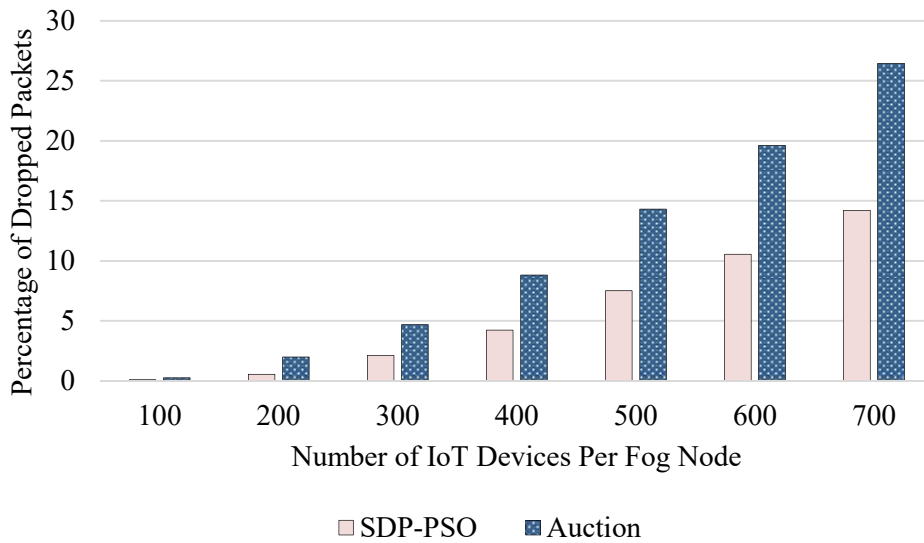
- Increasing the number of IoT devices per fog node, explicitly increases the number of channel demands. The queuing delay will increase as the packets will be queued longer time to be allocated a channel. The packets' chance to be dropped increases with the increase of their waiting time in the queue. In addition the fairness index decreases as it is a factor of both the channels (slots) demanded and allocated. As the demand increases and the possible allocation is the same (same available slots are used in all cases), so the fairness index decreases.



(a) Average Packet Queuing Delay



(b) Fairness Index



(c) Percentage of Dropped Packets

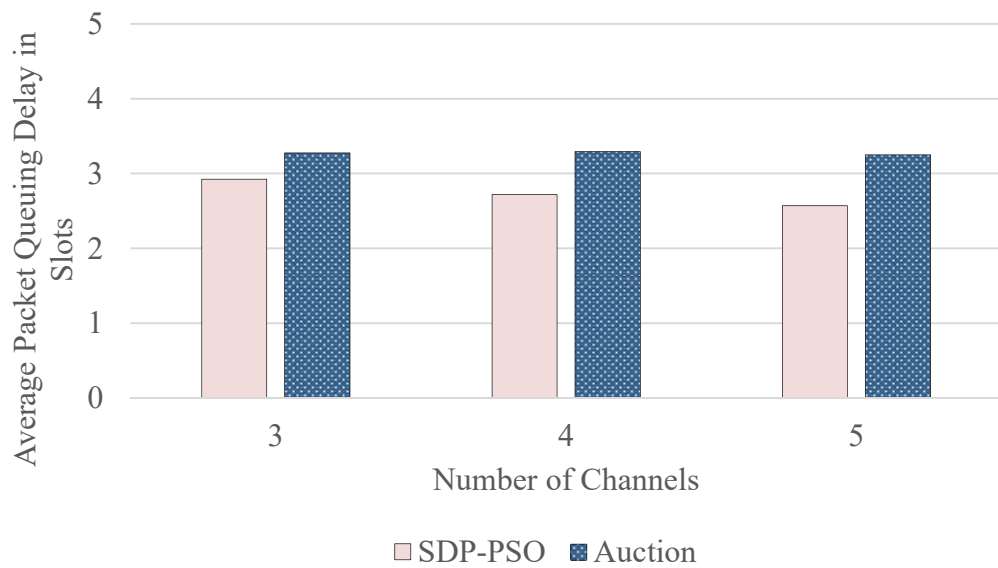
Figure 6.5 Performance Evaluations of the two Protocols Using Different Number of IoT devices per Fog Node

- By increasing the number of IoT devices per fog nodes, the measured metrics get worth, but not with the same rate in both protocols. Spectrum auction protocol performance decays by larger rate as it only allows allocating one channel per slot for each fog node.

**b. Changing number of channels.**

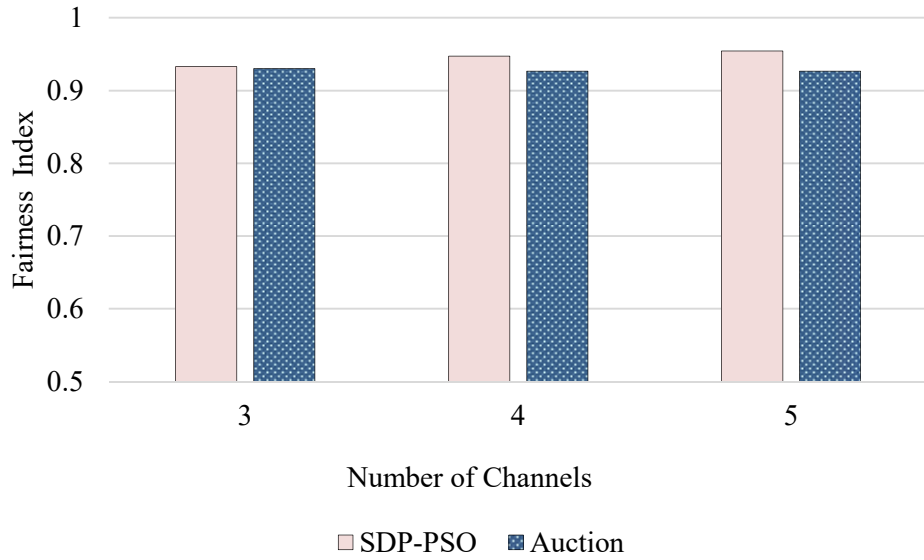
The graphs in Figure 6.6 illustrate the three evaluated metrics resulted when applying both protocols, using different number of channels, that varied for 3 to 5. The number of IoT devices per fog node and the interval by which each IoT device sends packets are kept constant with values 500 and 0.5 to 5 mins respectively. From these graphs, the following points are concluded:

- Increasing the number of used channels, will increase the number of available slots being used. This enhances the measured metrics of SDP-PSO protocol. As the channels assignments will be more (i.e. less dropped packets and higher efficiency) and faster (i.e. lower average packet queuing delay).
- The increase in number of channels won't affect the spectrum auction protocol by a noticeable value. As even there exist more available channel slots, the central server won't be able to assign more than one channel per fog node at each time slot.

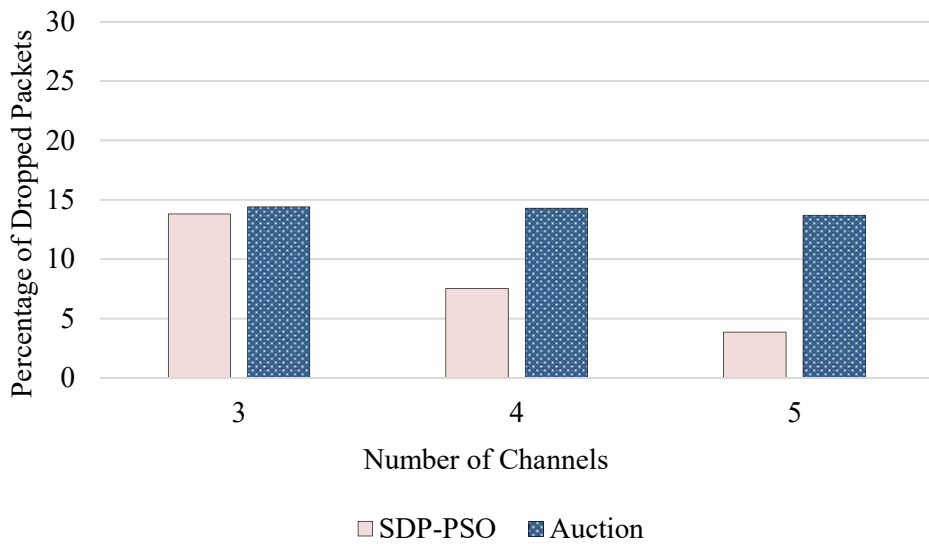


(a) Average Packet Queuing Delay





(b) Fairness Index



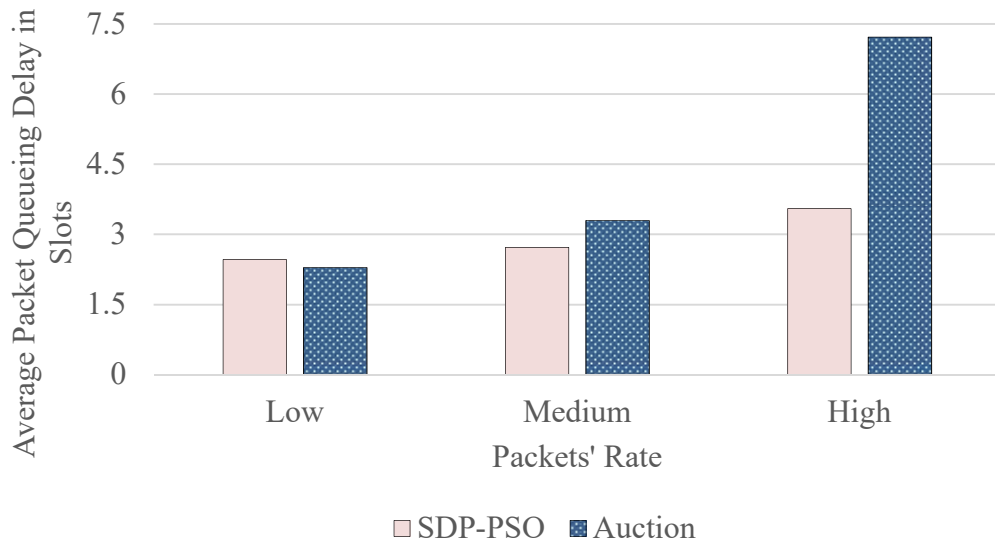
(c) Percentage of Dropped Packets

Figure 6.6 Performance Evaluations of the two Protocols Using Different Number of Channels

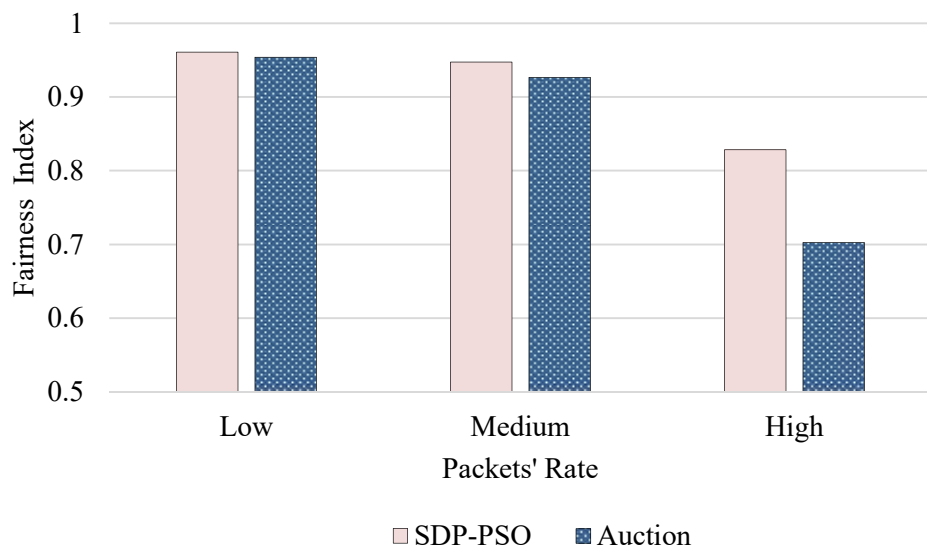
**c. Changing packets' rate.**

The graphs in Figure 6.7 illustrate the results of the evaluated metrics when applying both protocols, using various packets' rate. Changing the packets' rate is presented by changing the interval boundaries by which each IoT device sends packets ( $t_{min}$  and  $t_{max}$ ), these values are determined based on real medical sensors

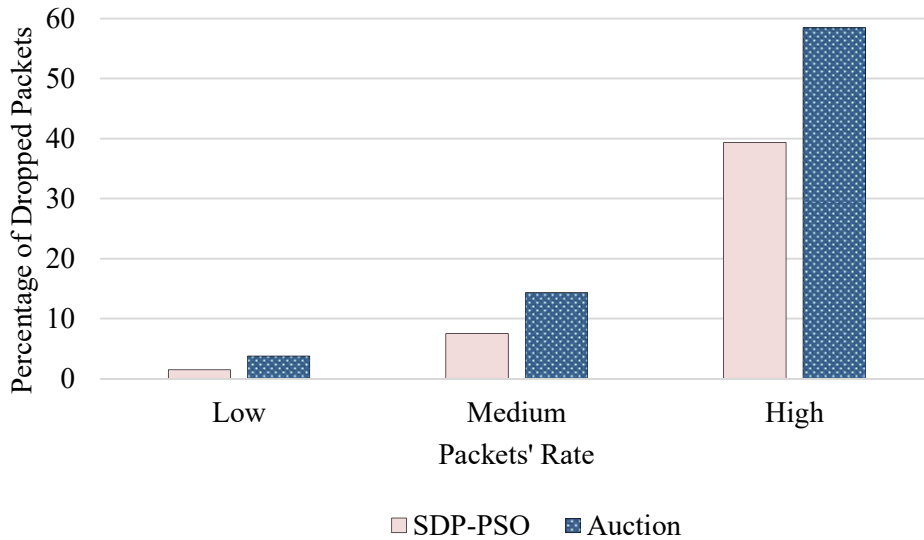
[109]–[111]. Intervals vary from 0.5 to 10 minutes for low packets rate, from 0.5 to 5 minutes for medium packets rate and from 0.5 to 1 minute for high packets rate. The number of IoT devices per fog node and the number of used channels are kept constant, with values 500 and 4 respectively. From these graphs, the following points are concluded:



(a) Average Packet Queuing Delay



(b) Fairness Index



(c) Percentage of Dropped Packets

Figure 6.7 Performance Evaluations of the two Protocols Using Different Packets' Rate

- Decreasing the packets' sending interval of the IoT devices leads to increasing the network's load, so increasing the channels' demands. Dealing with the same number of available channels, the queuing delay increases along with the packet dropping rate, while the fairness index among fog nodes decreases.
- The increase in the network load affects the spectrum auction protocol by a greater way, i.e. the spectrum auction evaluated metrics' degradations' rate is larger than that of the SDP-PSO protocol.

### 6.4.3 Results analysis

Table 6.3 summarizes the results of the proposed protocol and illustrates the percentage of improvement when using the SDP-PSO protocol compared to the spectrum auction protocol.

SDP-PSO protocol schedules the channels for a number of future time slots, which can be both an advantage and disadvantage. It is an advantage as it makes a near best channel allocation scheme that satisfies the three parameters' aim in the objective function, for all the slots available in this time interval. Also, scheduling for

a time interval gives an opportunity for the scheduler to apply the optimization technique on a number of time slots (hence greater number of available slots). But it can also be a disadvantage as the channel demands for a time interval will have to wait till the next time interval to be scheduled.

The main problem with the spectrum auction protocol is that it is mathematically designed to give only one channel per slot for each fog node, this restriction limits the usage of this protocol.

Table 6.3 Percentage of Improvement using the Proposed SDP-PSO Protocol Compared to the Spectrum Auction Protocol

Changed Parameter	Performance Metric	SDP-PSO	Auction	Average Percentage of improvement of SDP-PSO over Auction
Number of IoT devices	Average Packet Queuing Delay in Slots	2.633	2.892	8.93%
	Average Fairness Index	0.951	0.933	1.97%
	Average Percentage of Dropped Packets	5.616	10.882	48.40%
Number of Channels	Average Packet Queuing Delay in Slots	2.738	3.277	16.45%
	Average Fairness Index	0.945	0.928	1.81%
	Average Percentage of Dropped Packets	8.393	14.163	40.74%
Packets' Rate	Average Packet Queuing Delay in Slots	2.912	4.270	31.80%
	Average Packet Queuing Delay in Slots	0.912	0.861	5.90%
	Average Packet Queuing Delay in Slots	16.112	25.506	36.83%

Simulation results proved that SDP-PSO is better than the spectrum auction under different circumstances. It almost gives better results for every studied case, higher fairness, lower average queuing delay and lower number of dropped packets.

For the complexity, it has been proved that SDP-PSO protocol has lower time and space complexity than the spectrum auction protocol. It can also be used in real time unlike the spectrum auction protocol.

## **6.5 Conclusion**

This chapter addressed the problem of packets' scheduling for CRIoTs. It proposes a centralized time slotted algorithm that uses Discrete Permutation Particle Swarm Optimization (DP-PSO) for scheduling the packets every time interval (which consists of a number of slots). The protocol is named Scheduling based-on DP-PSO (SDP-PSO). An objective function is used with aim of maximizing the fairness index among fog nodes, minimizing the packets' queuing delay and minimizing the number of dropped unsent packets that exceeded their time in the network. A detailed sequence diagram and flow diagram were given for the proposed protocol.

In order to evaluate the proposed SDP-PSO, it is simulated along with an old technique named spectrum auction proposed in [95] using Matlab. Simulation results proved that the performance of the proposed SDP-PSO protocol overcomes that of the spectrum auction under different circumstances. It leads to better results in terms of the fairness, the average queuing delay and the number of dropped packets while varying the number of IoT devices per fog node, the number of channels and the packets' rate. It is also proved that the SDP-PSO protocol has lower time and space complexity than the spectrum auction technique. It can be used in real time in contrary to the spectrum auction.

The work in this is chapter is the second step out of the three steps of the work in this thesis. It proposed and proved the efficiency of a centralized packet scheduling technique based-on DP-PSO. The next chapter's purpose is to customize this scheduling strategy for applying it in distributed scheduling using a different IoT application.

**Chapter 7**

**Distributed Packets**

**Scheduling Technique for**

**Cognitive Radio Internet of**

**Things Based on Discrete**

**Permutation Particle Swarm**

**Optimization**

# Chapter 7

## **Distributed Packets Scheduling Technique for Cognitive Radio Internet of Things Based on Discrete Permutation Particle Swarm Optimization**

As mentioned all over the thesis merging the Cognitive Radio Networks with Internet of Things forming Cognitive Radio Internet of Things (CRIoTs) is an effective and economical solution to deal with the lack of spectrum problem. A new centralized packets' scheduling strategy for CRIoTs was proposed in chapter 6. Dealing with the scheduling problem as an optimization function, discrete permutation particle swarm optimization technique was used to solve this optimization problem. Using a health care facility as a case study, an objective function was formulated to maximize the fairness index among the fog nodes responsible for collecting and sending the IoT device's packets, minimize both the packets' queuing delay and the number of dropped packets at each time interval.

Some Cognitive Radio Internet of Things (CRIoTs) applications force the usage of distributed scheduling protocols, such as traffic management problem. This chapter studies how to use the scheduling protocol proposed in the previous chapter to be applied for distributed CRIoTs applications. Almost the same objective function is used but with some changes according to the nature of the used distributed application, taking the traffic management application as a case study.

The chapter is divided into four sections. A brief explanation on distributed scheduling is presented in section 1. Section 2 proposes using three distributed access strategies along with the SDP-PSO proposed in chapter 6, with aim to schedule the packets in a distributed manner. These strategies are evaluated in section 3. Finally section 4 concludes the facts obtained through the chapters.

## 7.1 Distributed Scheduling

IoT can be considered as a network of networks that allow a huge number of devices to communicate with each other. With the rapid increase in the IoT applications, deploying wireless partial mesh topologies that communicate in a distributed way will be more practical than centralized ones. Partial mesh topologies are highly flexible topologies, as it provides some redundant routes to ensure some level of fault tolerance. Wireless partial mesh topologies consist of several wireless nodes, which transfer data by acting as a forwarding node to its neighboring ones. Wireless partial mesh topologies are scalable and more flexible to work with more than one protocol, however partial mesh topologies are more complex where their nodes are directly and dynamically connected in a non-hierarchical manner.

Choosing an efficient and functional distributed/decentralized scheduling strategy for various wireless partial mesh topologies is a challenging task. Distributed scheduling strategy can be referred as; the sequence by which the network's nodes access and share the medium using only its information or may be the information of their neighbors. A distributed scheduling strategy is preferred as it is automatic, scalable and has no single point of failure. However it must maintain high throughput, low latency for both the set up phase and the transmission phase, also efficient energy consumption, low management overhead and other network services must be guaranteed. Several researches proposed schemes for wireless partial mesh topologies, various distributed scheduling schemes exist such as random access scheme, pseudo random scheme, cyclic/round robin access scheme, reservation, etc. [112]–[114].

Random access scheme, pseudo random scheme, cyclic/round robin access scheme and reservation scheme are the most general popular classification for wireless partial mesh topologies distributed schedulers. In these schemes, each node has a turn to apply the built protocol. Each one of this category is an umbrella under which several researches were done, also each one has its advantages, disadvantages and the applications that can operate by this scheme [112]–[115].

In random access scheme, there is no specific recurrence or duration for the networks nodes to access the medium, also the nodes can't predict neither their own nor their neighbors' communication period. Pseudo random scheme also hasn't any



specific recurrence or duration for the networks nodes to access the medium like random access scheme, however each node have the ability to predict the communication period for itself and also for its neighbors. For the cyclic access scheme; also called; round robin access scheme, there exist a fixed repetition and sequence pattern for the medium access that stays constant over a period of time. Reservation access scheme works by allowing the nodes having the intention to send to reserve future slots [114], [115].

## **7.2 Concept of the Proposed Distributed Scheduling**

As some IoT applications operate in a centralized manner, a new centralized packets' scheduling strategy for CRIoTs was proposed in the last chapter, it can be applied in several IoT centralized applications such as smart healthcare facility, home IoT application, IoT agriculture applications, etc. However some IoT applications do not operate in a centralized manner completely, instead they operate in a centralized way in some layers and in a distributed way in other layers, such as smart traffic, energy consumption, etc. So a new distributed scheduling strategy must be proposed for these layers to operate in a distributed manner in the IoT applications.

IoT smart traffic application is an active research point and a great percentage of the IoT researches were done for developing this application; some cities already deploy smart traffic control, note that this application needs a considerable network bandwidth. We considered using this application when applying our new proposed distributed scheduling strategy.

Considering an IoT network to be a wireless network with mesh topology, a distributing scheduler must exist to organize the access to the shared medium. In this chapter we propose using three access schemes, the classical round robin and two proposed ones named vertex cover and enhanced version of round robin, to arrange the nodes' sequence that performs the scheduling upon the estimated free slots resulted from chapter 5 by applying the SDP-PSO proposed in chapter 6.

## 7.2.1 System Structure

Apply the IoT smart traffic application, there are several architectures that were proposed for it [116]–[118], but most of them are approximately similar. In general, the traffic's system architecture is hierarchical, it is composed of several levels as

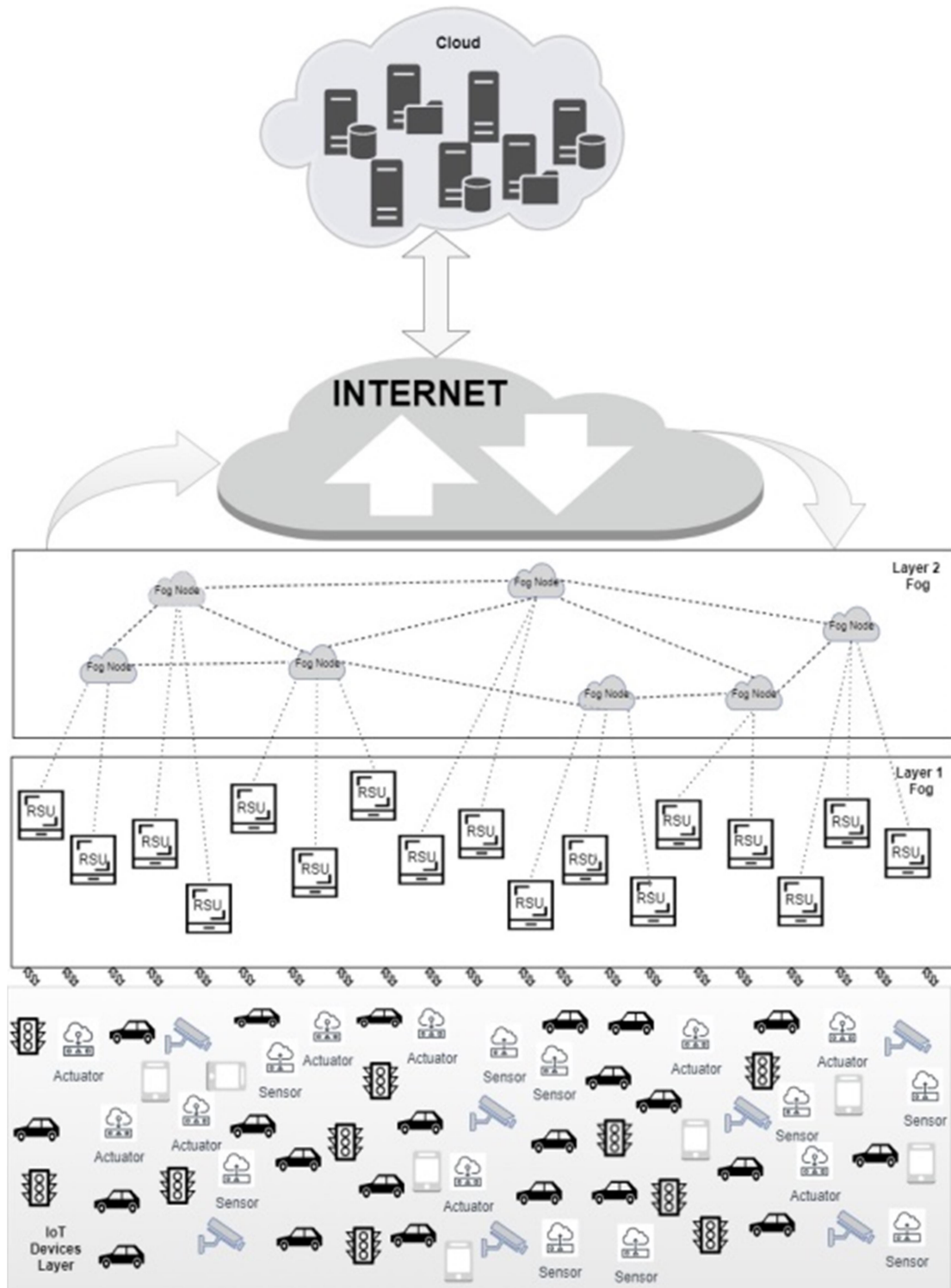


Figure 7.1 System Architecture

shown in figure 7.1; IoT devices which are sensor and actuators installed in several things such as vehicles, traffic light signal cameras, etc. Road Side Units (RSUs) which can be considered as the first fog layer that provides the link and communication between the IoT devices and the second layer fog/edge nodes. The second fog layer that performs several tasks, but the two most important ones our research concerned with are the scheduling and data analysis. Fog computing is not a replacement for cloud computing, so, certainly the final level of the architecture must include cloud nodes which perform powerful processes. Each of these layer is explained in detail as follows: [119], [120].

- ***IoT Layer***; this layer consists of several kinds of devices (things) but mainly it can be considered as embedded sensors and actuators. Sensors are used to measure various parameters such as vehicles speeds, carbon dioxide level, roadway temperatures, number of cars waiting at an intersection, etc. Actuators are used to regulate several things such as idle speed, control air flaps for torque, power consumption optimization, etc. Other devices are used to collect and transmit data such as cameras. These IoT devices transmit data to the layer above, the RSUs layer.
- ***RSUs Layer (First Fog Layer)***; this layer consists of several RSUs that are installed alongside roads. They are used as wireless access points that provide communication to the IoT devices within their wireless coverage. The RSUs gather the data from the IoT layer, then packetize and prioritize it, after that it sends the packet schedule's requests to the above fog layer. Also it forms acknowledgements for the received emergency packet (level five priority) from the second fog layer.
- ***Second Fog Layer***; this layer consists of several fog nodes, each fog node is connected to several RSUs in the first fog layer. It is also connected to some fog nodes in the same layer, to form a partial mesh topology network, as shown in figure 7.1. This fog node layer is responsible for several tasks mainly scheduling the packet requests received from the RSUs, analyzing the received data and forming their own packets based on this analysis.
- ***Cloud Layer***: The cloud layer consists of several servers and data storage. This layer performs powerful tasks such as big data processing, data warehousing and

service management.

Our work is concerned with scheduling packets transmission between RSUs and the second fog layer.

## 7.2.2 Problem Formulation

The protocol proposed in this chapter is the final step in our research. In chapter 6 a new centralized time slotted algorithm was proposed. It uses discrete permutation particle swarm optimization for scheduling packets among predicted primary network' empty time slots. An objective function was formulated to maximize the fairness index among fog nodes, minimize the packets' queuing delay and the number of dropped packets at each time interval.

Applying IoT smart traffic application; this chapter proposes a distributed strategy for scheduling packets transmission between RSUs and the second fog layer, among the predicted free slots obtained by applying any CRNs' channel status prediction technique, for example the 2-model HMM-SVM protocol proposed in chapter 5. The fog nodes in the second fog layer will take turn to act as a central node to perform the scheduling process using the centralized time slotted protocol that uses discrete permutation particle swarm optimization proposed in chapter 6. An objective fitness function near similar to that used for the centralized protocol in equation 6.16 is applied using discrete-permutation particle swarm optimization but with some differences according to the distributed scheduling scheme needed to be developed. The objective fitness function combines fairness among RSUs, the packets' queuing delay and the number of dropped packets from those needed to be scheduled at the fog node responsible for performing the scheduling process at such time interval. This subsection is divided into two parts; the first part presents the mathematical formulation of the problem, while the second one explains the problem in details.

### A. Mathematical Formulation

Table 7.1 illustrates the variables' notation used while formulating the problem mathematically. It is divided into two parts; part (a) states the variables' notation changed from last chapter, while part (b) is the unchanged ones.

Table 7.1 Variables' Notation  
(a) Changed Variables' notation

<b>Parameters</b>	<b>Parameters Definition</b>
$FR_{p(f)}(T)$	Fairness index of particle p at fog node f per RSU at interval T.
$Q_{p(f)}(T)$	Queuing delay of particle p at fog node f in interval T.
$D_{p(f)}(T)$	Number of dropped packets of particle p at fog node f in interval T.
$R_{NR}$	Number of RSUs
$aR_j(T)$	Number of allocated slots for RSUj in interval T.
$dR_j(T)$	Number of demanded slots by RSUj from previous intervals that weren't scheduled.
$N_{pf}(T)$	Number of packets in particle p at fog node f to be sent in interval T.
$NS_{f(Pr)}(T)$	Number of packets at fog node f with priority Pr that weren't scheduled in the interval T.
$P_{Ri,j,RT}$	Packet request generated by RSUi, belonging to fog node j requested at time $R_T$
$o(P_{Ri,j,RT})$	Order of $P_{Ri,j,RT}$ in the waiting queue of the fog node in turn.

(b) Unchanged Variables' notation

<b>Parameters</b>	<b>Parameters Definition</b>
$T.$	Previous intervals prior to T.
$AT_i$	Allocation time of packet i.
$RT_i$	Request time of packet i.
$Prior_i$	Priority of packet i.
$Pr$	Priority level.
$P_{ES}$	Predicted empty slot.
$L$	Interval size in slots
$NP_{ES}$	Number of predicted empty slots.
$C$	Number of channels.
$T$	Time interval, consists of a number of successive slots.
$P_{max_i}$	Number of slots after which the packet will be dropped if not scheduled.

**a) Objectives**

$$\begin{cases} \text{Maximize } (FR_{p(f)}(T)) \\ \text{Minimize } (Q_{p(f)}(T)) \\ \text{Minimize } (D_{p(f)}(T)) \end{cases} \quad (7.1)$$

*where*  $FR_{p(f)}(T) = \frac{(\sum_j X_j)^2}{R_{NR(T-)} \sum_j X_j^2}$  ,  $X_j = \frac{aR_j(T)}{dR_j(T-)}$  (7.2)

$$Q_{p(f)}(T) = \sum_{i=1 \rightarrow N_{p_f}(T)} (AT_i - RT_i) * Prior_i \quad (7.3)$$

$$D_{p(f)}(T) = \sum_{Pr=1 \rightarrow \text{Highest priority}} \sum_{NS_{f(Pr)}} Pr * NS_{f(Pr)}(T) \quad (7.4)$$

**b) Decision Variables**

$$Pairs_{(f)} (P_{R_{i,j}, RT}, P_{ES})_T$$

**c) Constrains**

$$P_{max_i} < L * Prior_i \quad (7.5)$$

$$NP_{ES} \leq C * L \quad (7.6)$$

$$P_{ES} = \text{null if } \forall j dR_j = 0 \quad (7.7)$$

$$\exists! P_{R_{i,j}, RT} \in (P_{R_{i,j}, RT}, P_{ES}) \ \& \ \exists! P_{ES} \in (P_{R_{i,j}, RT}, P_{ES}) \quad (7.8)$$

$$\text{If } Prior_i = \text{highest, } o(P_{R_{i,j}, RT}) = 1 \quad (7.9)$$

Note that the equations 7.1 to 7.9 are near similar to the equations 6.7 to 6.15 used in chapter 6, however their substitution is changed according to their appliance on the fog node responsible to schedule at the time interval T.

**B. Problem Explanation**

This part explains in details the problem formulated in the previous one. In our system there exist a number of fog nodes, each with a connected number of RSUs  $R_{NR}$ , each RSU is connected to a number of IoT devices. The predicted empty slots  $NP_{ES}$  are obtained by applying the 2-model HMM-SVM protocol proposed in chapter 5, they are used in the allocation process. Each empty slot is divided into successive intervals T with L time slots.

Each node has a turn to apply the built protocol, i.e. at each time interval  $T$ , the fog node in turn will receive several packets' requests  $P_{R_{i,j},RT}$ . Each packet is assigned a priority ( $Prior_i$ ) by its RSU, according to its abnormality. The priority of each packet  $i$  will determine the maximum time  $Pmax_i$ , it can be queued before being dropped if it wasn't allocated a channel.

Each fog node in its turn, deals with the scheduling problem as an optimization one, using the fitness function in equation (7.10). This fitness is near similar to the equation (6.16) in chapter 6, but with some changes as the SDP-PSO is applied in a distributed way. Three metrics were included in this function; the fairness index among the RSUs requesting channels, the packets' queuing delay and the number of dropped packets (those exceeding their maximum time waiting in the RSUs buffers without being scheduled).

**a) Objectives**

The used three objectives (stated in equations 7.2, 7.3 and 7.4) are somehow similar to the objectives used in the last chapter, except that the fairness index is calculated among the RSUs requesting channels rather than among the fog nodes. Also the considered packets are those of only the fog node responsible for scheduling at such time interval. The used three objectives are to be optimized at each time interval ( $T$ ) by the fog node in turn to perform the scheduling process.

Weighted sum method was used to solve the multi objective function for simplicity and rapidity. All the metrics in equations 7.2, 7.3 and 7.4 used in the multi objective function are united into one scalar. The multi objective function in equation 7.1 is transferred into the following single objective optimization function:

$$fitness(T) = w_1 * FR_{p(f)}(T) + w_2 * (-Q_{p(f)}(T)) + w_3 * (-D_{p(f)}(T))$$

$$where: \quad w_1 + w_2 + w_3 = 1 \quad (7.10)$$

As in chapter 6; positive sign is assigned to the fairness in order to maximize it, while negative sign is used for the queuing delay and the number of dropped packets in order to minimize them. The first metric takes values from 0 to 1, so it is kept as it is, while the two other metrics can have any value, so they are normalized (to get a value from 0 to 1) before using them.

### ***b) Decision Variables***

The decision variables are similar to the one in chapter 6, but it is produced by the fog node in turn to perform the scheduling process. At each time interval T, the allocation schedule produced a number of pairs as follow:

$$Pairs_{(f)} (P_{Ri,j,RT}, P_{ES})_T$$

### ***c) Constrains***

The constrains from 7.5 to 7.9 are near similar to those stated in equations 6.5 to 6.9 in the last chapter, but applied only on the packets at the fog node (f) responsible for the scheduling process at interval T, not all the packets in the system.

## **7.2.3 Proposed Distributed Medium Access**

Our research is concerned with packet scheduling between the RSUs (first fog layer) layer and the second fog layer. It is performed by the fog nodes in the second fog layer. Dealing with a shared medium, a distributed scheduling strategy – for the wireless partial mesh topology formed between the fog nodes – must exist to control the share and the access to the medium. This chapter proposes using three strategies for controlling the access and the usage of the estimated available channels resulted applying the protocol proposed in chapter 5. The three strategies specify the sequence by which the fog nodes will apply the SDP-PSO scheduling protocol proposed in chapter 6 to schedule the packet requests sent by the RSUs. The three access strategies are; the Classical Round Robin (RR), the Enhanced Round Robin (ERR) and the Vertex Cover (VC). But first some assumptions must be illustrated for the scheduling between the two layers.

### ***System Assumptions***

- Fog nodes are considered high capacity wireless gateways.
- The topology of fog nodes is considered to be wireless partial mesh topology. In this topology, fog nodes are distributed over a geographical area, such that each fog node has various number of neighboring nodes, which represents a subset from the total number of fog nodes in the network.
- Each fog node is connected to various numbers of Road Side Units (RSUs).



- Each RSU periodically sends its packets' request to be scheduled to the fog node connected to it.
- Channels reuse is not considered, i.e. all fog nodes compete on the same operating frequencies, which is divided into channels.
- The number of network fog nodes is known by all the fog nodes.
- Each fog node has a unique Id (MAC Address), which is known to all other fog nodes in the network.
- Each fog node knows at least its 1-hop neighboring fog nodes.
- At the network setup phase the order by which the fog nodes in turn will be in charge of performing the scheduling are specified using messages. This setup phase is done periodically or when a change occurs in the network topology.
- The fog node in turn uses the predicted empty slots resulted from applying the protocol proposed in chapter 5.
- The fog node in turn uses SDP-PSO to form a generation that consists of a number of particles used to schedule a number of packets.
- The produced particles include a number of pairs consisting of the scheduled packet's number and the predicted empty slot allocated to it.
- For each particle in the generation, the three parameters; the fairness index, the queuing delay and the number of dropped packets, used in the objective function equation (7.10); are measured, in order to conclude a near best allocation pattern derived after several iterations.

The following paragraphs explain the three used access strategies. The term "cycle" used all over the rest of the chapter, refers to a number of intervals  $T$ , each composed of a number of slots sent by a specific fog node following a certain order. Note that the term "cycle" is virtual, as not all the cycle intervals must be sent together, instead each interval allocation composed by a fog node is sent the specific RSUs to start sending.

## A. Classical Round Robin

In classical round robin the fog nodes are arranged using a predefined order. According to its arrangement in a cycle, each fog node in the network takes its turn to apply the SDP-PSO for the packets' requests formed by itself (which are the emergency packets acknowledgement) and the packets' request sent by the RSUs connected to it. The time interval of the classical cycle is only one slot, i.e. the fog node in turn can use all the predicted free slots in the channels in each time interval which is considered to be to one slot. Each fog node in its turn choose the particle, that represents the near best solution resulted from solving objective function represented in equation 7.10 by applying SDP-PSO, among packets' request coming from the RSUs connected to this node and also the priority five packets' acknowledgment formed by the node itself.

Figure 7.2 illustrates an example of using the classical round robin together with the SDP-PSO as distributed scheduling for CRIoT. The networks is composed of eight nodes, each node sends a particle in its turn, so the cycle is composed of the particle of Fog node 1, then that of Fog node 2, etc., until the particle of Fog node 8. For example the particle of fog node 2, is the near best solution resulted from applying SDP-PSO among packets' request coming from the RSUs connected to fog

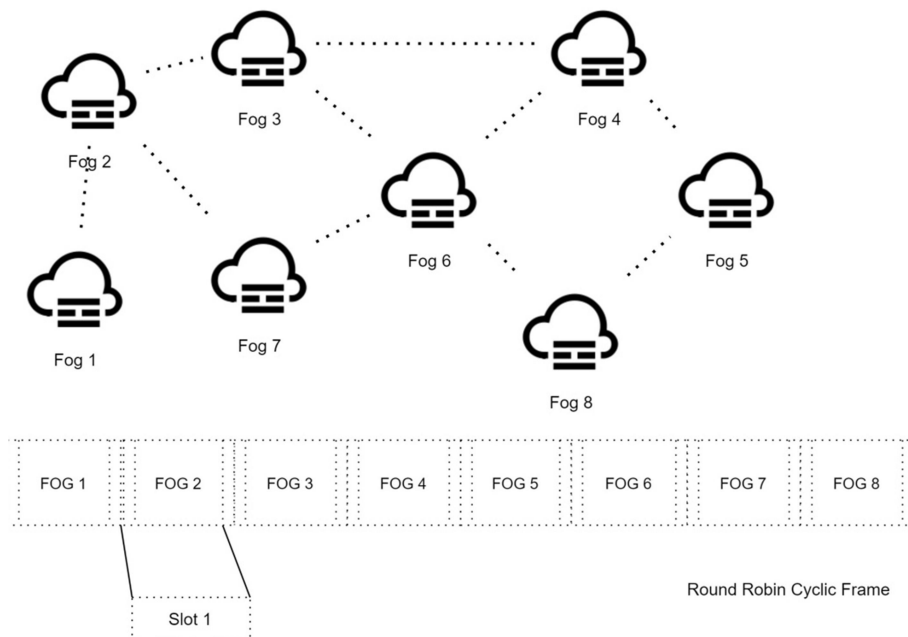


Figure 7.2 Classical Round Robin Cycle

node 2 and also the priority five packets' acknowledgment formed by fog node 2 itself.

### B. Enhanced Round Robin

In the proposed enhanced round robin the fog nodes are arranged using a predefined order. Each fog node periodically sends its packets schedule requests (either its packets or those coming from its RSUs) to its neighboring fog nodes. Each fog node acquires its right to use the estimated available channels in the interval according to its arrangement in the cycle. It forms a schedule that consists of several slots that schedule the packets' request formed by itself, and those sent by its connected RSUs and also the packets' request sent by its neighboring fog nodes. In other words; each fog node in its turn acts as the cluster head for its neighboring nodes, it acquires the right to schedule its and their packets requests. The fog node in turn solves the objective function represented in equation 7.10 by applying SDP-PSO, to choose the near best solution (particle) among packets' request formed by the fog node or sent by its RSUs or sent by other fog nodes in its neighborhood.

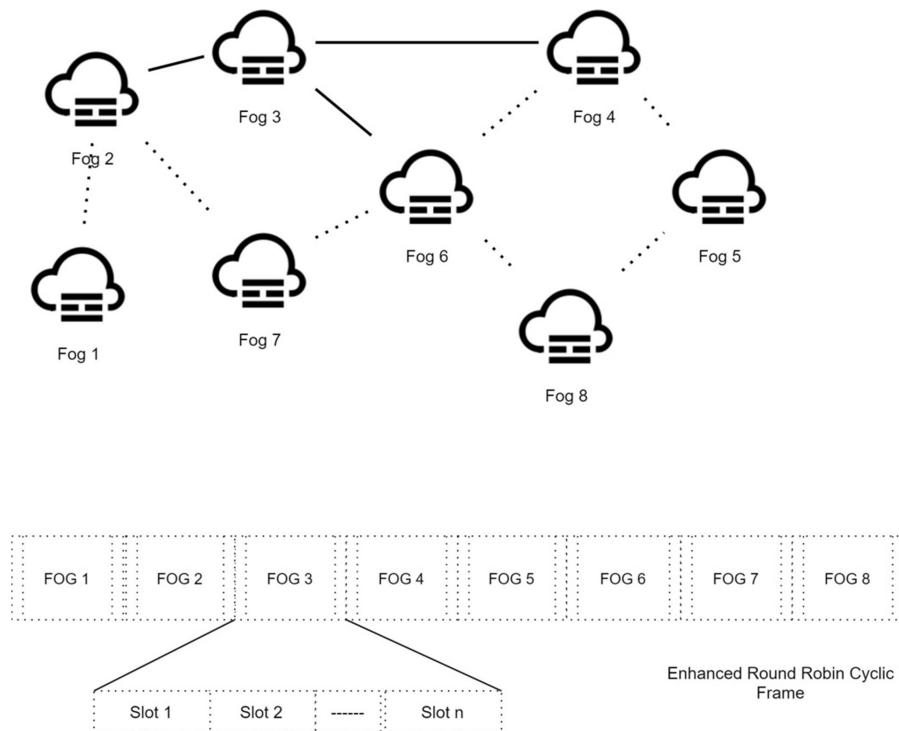


Figure 7.3 Enhanced Round Robin Cycle

Figure 7.3 illustrates an example of using the proposed enhanced round robin access strategy together with the SDP-PSO as a distributed scheduling for CRIoTs. The network is composed of eight nodes, each node sends a particle in its turn, so the cycle is composed of the particle of Fog node1, then that of Fog node2, etc., until the particle of Fog node8. For example the particle of fog node 3, is the near best solution resulted from solving the objective function represented in equation 7.10 by applying SDP-PSO among the packets' request coming from the RSUs connected to fog node 2, the priority five packets' acknowledgment formed by fog node 2 itself, in addition to the packets' request coming from the fog node connected to (seen by) fog node 3, i.e. fog nodes 2, 4 and 6.

### C. Vertex Cover

The vertex cover nodes are defined as the minimum number of nodes that covers all other nodes in a partial mesh network so that all fog nodes will be connected to at least one vertex node. This minimum vertex cover is an optimization NP- complete problem, where several researches were done for the approximation algorithms to find approximate solution in a reasonable time [121], [122].

In the set up phase the network applies a minimum vertex cover discovery protocol, to find the minimum number of fog nodes called vertex nodes that covers all the fog nodes in the partial mesh network so that all the fog nodes will be connected to at least one vertex node. The cycle will be divided among only the nodes chosen to be the vertex nodes. Each fog node periodically sends the packets schedule requests to the vertex fog nodes connected to it. Then, only the vertex fog nodes take turns to send. Each vertex node in its turn will solve the objective function represented in equation 7.10 by applying the SDP-PSO to schedule its own packet request, the packets' request sent by its RSUs and the packets requests received from its neighboring fog nodes.

Figure 7.4 illustrates an example of using the proposed vertex cover access strategy together with the SDP-PSO as a distributed scheduling for CRIoTs. The network is composed of eight nodes, but only three nodes 2, 4 and 6 are chosen to be the network vertices, they cover all the fog nodes so that each will be connected to at least one vertex node. Each of these vertices sends a particle in its turn, so the cycle is composed of the particle of Fog node 2, then that of Fog node 4 and finally

that of Fog node 6. For example the particle of fog node 6, is the near best solution resulted from solving the objective function represented in equation 7.10 by applying SDP-PSO among the packets' request coming from the RSUs connected to fog node 6, the priority five packets' acknowledgment formed by fog node 6 itself, in addition to the packets' request coming from the fog node connected to (seen by) fog node 6, i.e. fog nodes 3, 4, 7 and 8.

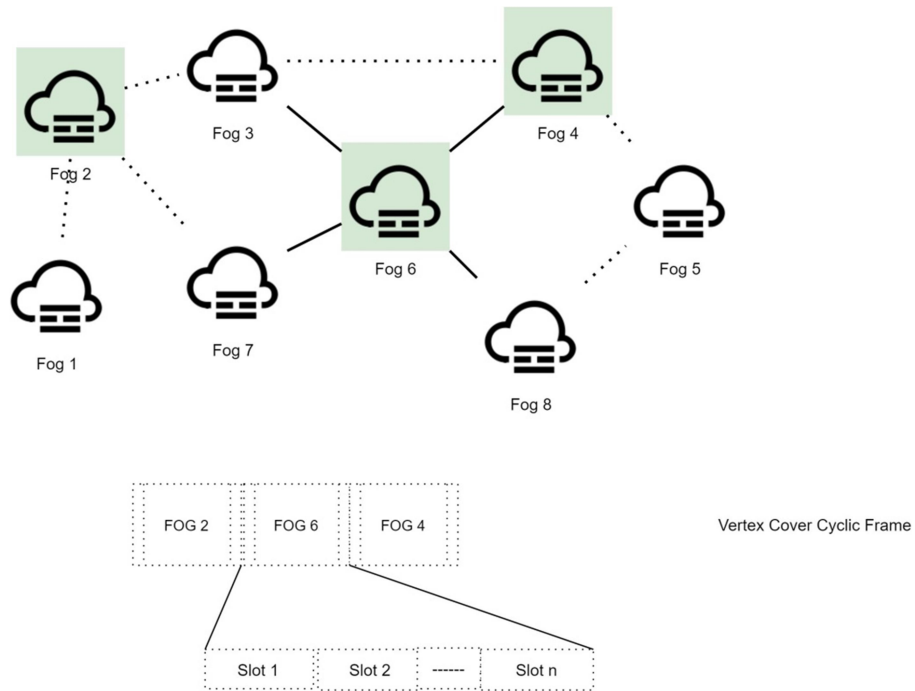


Figure 7.4 Vertex Cover Cycle

### 7.3 Performance Evaluation

In this section, the effect of combining each of the three access strategies, the classical round robin and the two proposed ones; the vertex cover and the enhanced round robin, with applying the SDP-PSO protocol proposed in chapter 6, to produce a distributed scheduling method for CRIOts is examined. This section consists of three subsections; the first and second one state the used performance metrics and the simulation parameters respectively. The third part illustrates and analyses the obtained simulation results.

### 7.3.1 Performance Evaluation Metrics

The three performance metrics used for the evaluation of the proposed distributed packets' scheduling strategies for CRIoTs are the fairness index among RSUs, the average packet queuing delay and the percentage of dropped packets. The fairness is calculated among the RSUs rather than the fog nodes. Also the considered packets are those waiting in the fog node queue responsible for the scheduling process at such time interval rather than all waiting packets.

#### A. Fairness index

Fairness index parameter is used to determine whether various RSUs are receiving a fair share of slots according to their demands.

$$FR = \frac{(\sum_j X_j)^2}{R_{NR} \sum_j X_j^2} \quad (7.11)$$

$$\text{where } X_j = \frac{ARS_j}{DRS_j}$$

FR is the fairness index among all the RSUs.

ARS<sub>j</sub> is the total number of allocated slots for RSU j.

DRS<sub>j</sub> is the total number of demanded slots for RSU j.

R<sub>NR</sub> is the total number of RSUs.

In order to allow RSUs to obtain fair shares of the estimated free slots, high fairness index must be obtained.

#### B. Average Packet Queuing delay

Average packet queuing delay is defined as the average time each packet stay in the fog node queue before being sent. Note that the concerned nodes from which the packets are coming differ from an access strategy to another one.

$$Q = \frac{\sum_{i=1 \rightarrow NR} (AT_i - RT_i)}{NR} \quad (7.12)$$

where Q is the average packet queuing delay

NR is the total number of received packets

AT<sub>i</sub> is the time the packet i is allocated a slot

RT<sub>i</sub> is the time the packet i requests for a slot.

In order to maintain an acceptable waiting time for the packets, the average queuing delay for the packets must be as low as possible.

### C. Percentage of Dropped Packets

Ordinary packets, if not sent; are discarded after remaining in the network for a predefined time for each priority. This time is variable according to packet priority.

$$D = \frac{ND}{N} * 100 \quad (7.13)$$

where D is the percentage of dropped packets.

ND is the total number of dropped packets.

N is the total number of packets.

In order to obtain good performance, the number of dropped packets must be minimized.

## 7.3.2 Simulation Parameters

The following simulation parameters are used:

- Number of slots: represents the used number of estimated slots resulting from applying the protocol proposed in chapter 5. Note that, the slots are not empty all the time, instead the primary users may be using it.
- Number of fog nodes: represents the number of fog nodes used in the architecture.
- Number of RSUs per fog node,  $R_{NR}$ : represents the number of Road Side Units per fog node.
- Number of channels, C: represents the number of used channels.
- Interval by which RSUs send packets: to indicate the packets' data rate.
- Number of slots after which the unsent packet will be dropped, Pmax: represents the number of slots the packets will be dropped if not sent. It varies depending on the packet priority. It is calculated by multiplying the packet priority with the number of slots in the time interval. Note that as the packet priority increases the packet will stay longer before being dropped.

- Interval size,  $l$ ; represents the interval size in slots.
- Network Connectivity,  $NC$ : represents the partial mesh network topology connection percentage.

Table 7.2 illustrates the values of the parameters used in the simulation.

Table 7.2 Simulation Parameters' Value

Parameter	Used Value(s)
Number of Slots	18000
Number of fog nodes	Varied from 4 to 7
Number of RSU devices per fog node	Varied from 4 to 10
Number of channels, $C$	5
Interval by which RSU devices send packets	from 1 to 20 slots Corresponding to 0.2 to 4 sec
Number of slots after which the packet will be dropped, $P_{max}$	Packet priority * interval size
Interval size in slots, $l$	2
Network Connectivity $NC$	Varied from 35% to 70%

### 7.3.3 Simulation Results

To study the performance of the three access strategies; Vertex Cover (VC), Enhanced Round Robin (ERR) and Classical Round Robin (RR); the three performance evaluation metrics calculated using equations 7.11, 7.12 and 7.13 are measured using the three access strategies in addition to the SDP-PSO protocol proposed in chapter 6. For the simulations, the number of RSUs per each fog node, the number of fog nodes and the network connectivity are changed in order to study the impact of their change on the performance, while keeping the other parameters constant.

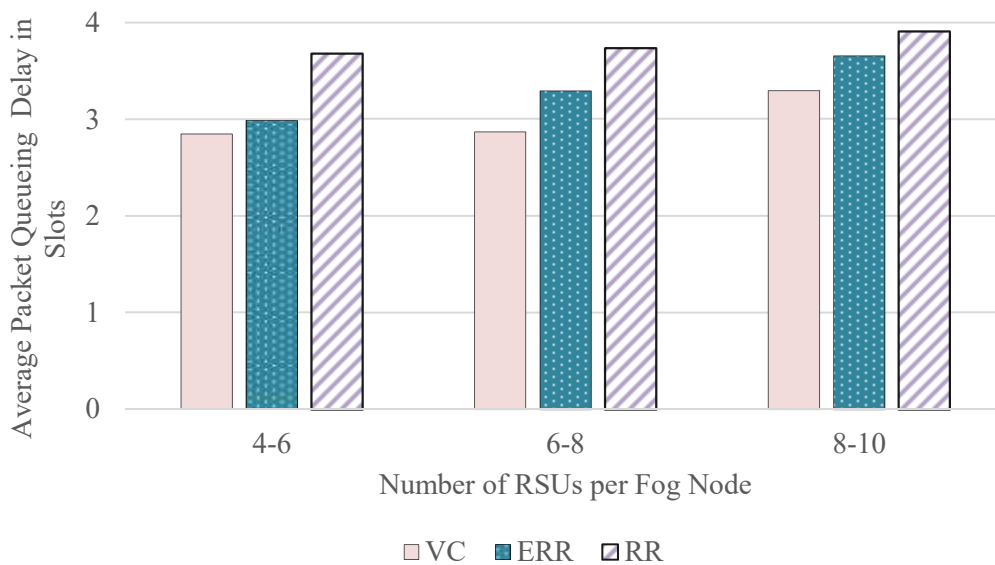
#### A. Changing number of RSUs per fog node.

The graphs in Figure 7.5 illustrate the three evaluated metrics resulted when applying the three distributed access strategies together with the SDP-PSO proposed

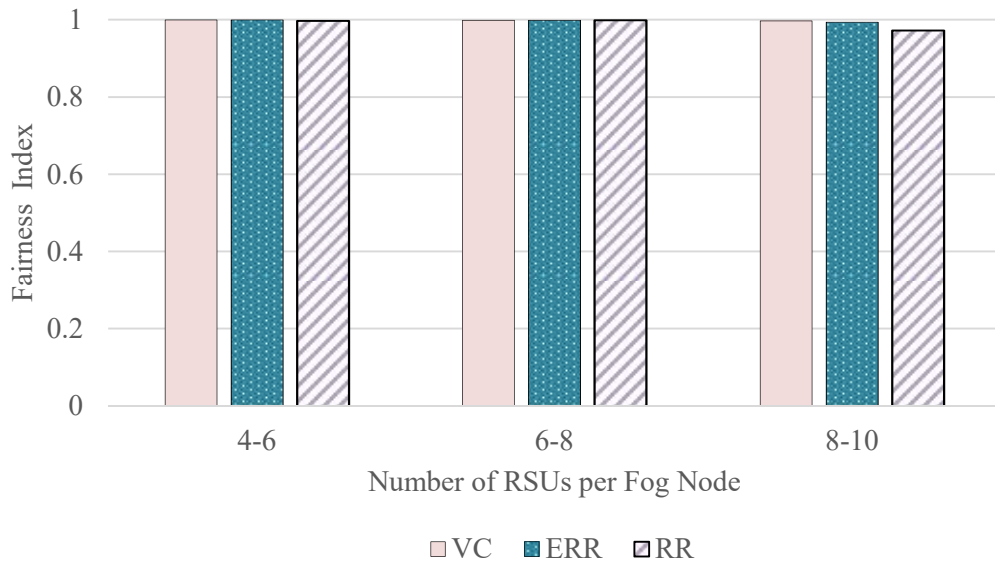


protocol in chapter 6, using various number of RSUs per fog node, that vary for 4-6 to 8-10. Note that a-b RSUs, means that the number of RSUs per fog node is randomly chosen between a and b. The number of fog nodes and fog nodes' network connectivity are kept constant, with values 6 and 50% respectively. The following points are the conclusion driven from these graphs:

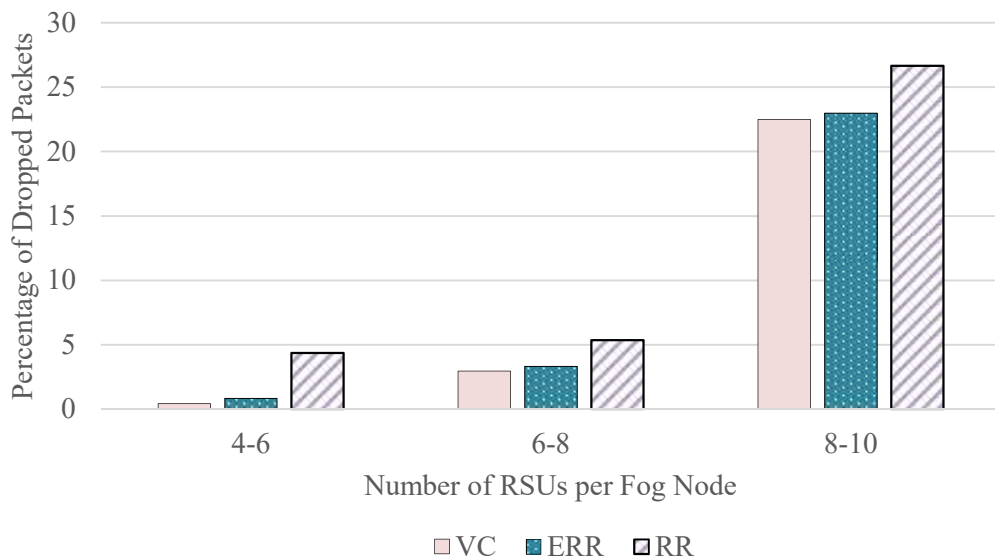
- All the three evaluated metrics will degrade while increasing the number of RSUs per fog node. This is logical as the increase of number of RSUs per fog node leads to the increase in the network load, so the increase of the channel's demands, while dealing with the same number of free channels slots, the packets will have to wait longer in the queue to be scheduled. This will result an increase in the average queuing delay, which consequently may lead to an increase in the percentage of dropped packets as the packets may exceed its time ( $P_{max}$ ) in the system, and they will have to be dropped.
- However the evaluated metrics' degradation doesn't have the same rate. The performance of the vertex cover access strategy degraded with a smaller rate as its cycle consists of only the vertex nodes' intervals, i.e. it is shorter than that of the two other strategies, so the nodes won't have to wait a long time to have their turn to schedule the packets' request they have. Also the vertex nodes can schedule not only their own packets and their RSUs' packets requests but also their fog node neighbors packets requests.



(a) Average Packet Queuing Delay



(b) Fairness Index



(c) Percentage of Dropped Packets

Figure 7.5 Performance Evaluations of the three Access Strategies Using Different Number of RSUs per Fog Node

- Both the cycle of the classical round robin and the enhanced round robin includes an interval for each fog nodes in the network, so each fog node have to wait all the other fog nodes in the network to have its turn to schedule the packet requests it has, so the evaluated metrics is worth than the vertex cover. However as the fog nodes in the enhanced round robin can schedule the packet requests of itself, its

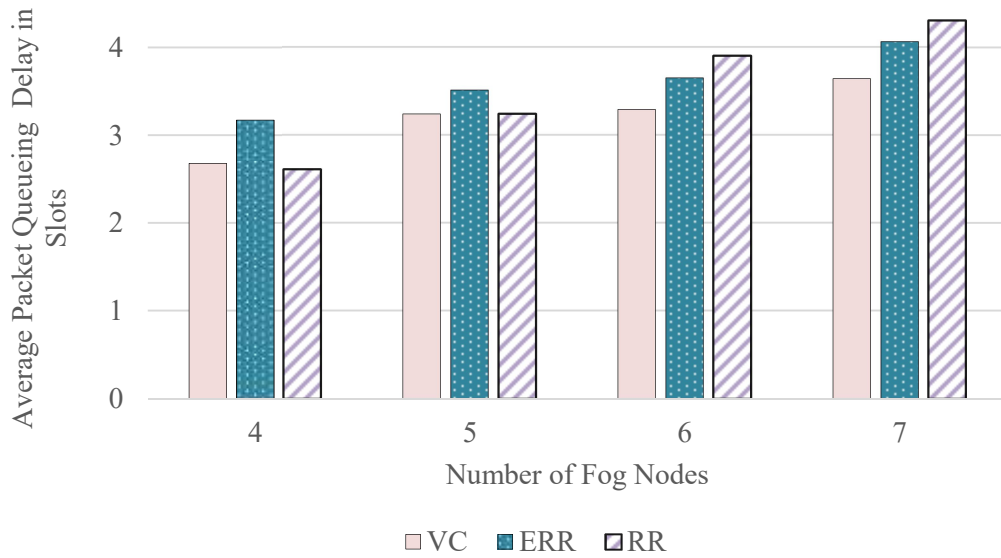
RSUs and that's of their neighbors RSUs so the evaluated metrics is better than the classical round robin which can only schedule the packet requests of itself and its RSUs.

## **B. Changing number of fog nodes**

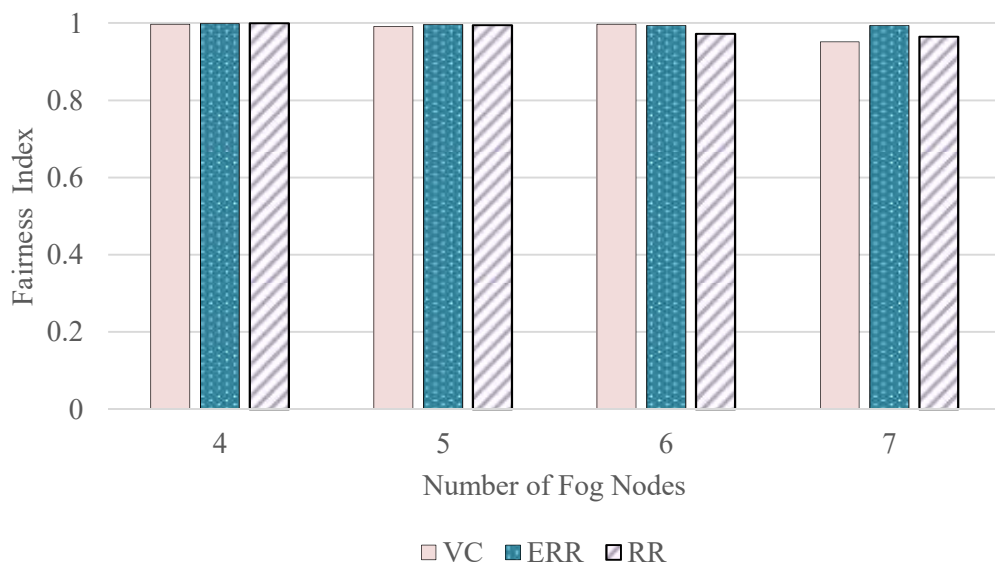
The graphs in Figure 7.6 illustrate the three evaluated metrics resulted when applying the three access strategies together with the SDP-PSO protocol proposed in chapter 6, using various number of fog nodes, that vary for 4 to 7. The number of RSUs per fog nodes and the fog nodes network percentage connectivity are kept constant, with values 8-10 and 50% respectively. The following points are the conclusion driven from these graphs:

- Dealing with the same number of RSUs, same network connectivity and same number of free slots, two factors affect the performance of the three access methods; the length of the cycle and the number of nodes whose packet requests are allowed to be scheduled at each time interval.
- Having a small number of fog nodes the vertex cover performance is near that of the classical round robin, but it is better than the enhanced round robin. The reason for this is that the cycle length (in terms of slots) of both strategies is nearly equal in this case. However the cycle of the enhanced round robin is approximately twice that of the other access strategies, so dealing with the same network connectivity 50% this strategy performance is not good when having a small number of fog nodes.
- Of course the performance of the three access strategies degrades with the increase of the number of fog nodes, as the total number of RSUs in the networks, and consequently the number of needed slots increases, while having the same number of empty slots.
- However in both the vertex cover and the enhanced round robin access strategies, increasing the number of fog nodes consequently increases the number of the fog nodes whose packet requests can be scheduled by others at each interval. This minimizes the effect of the problem arising from increasing the number of fog nodes (which increases the number of needed slots), i.e. their performance's degradation will be with an acceptable rate.

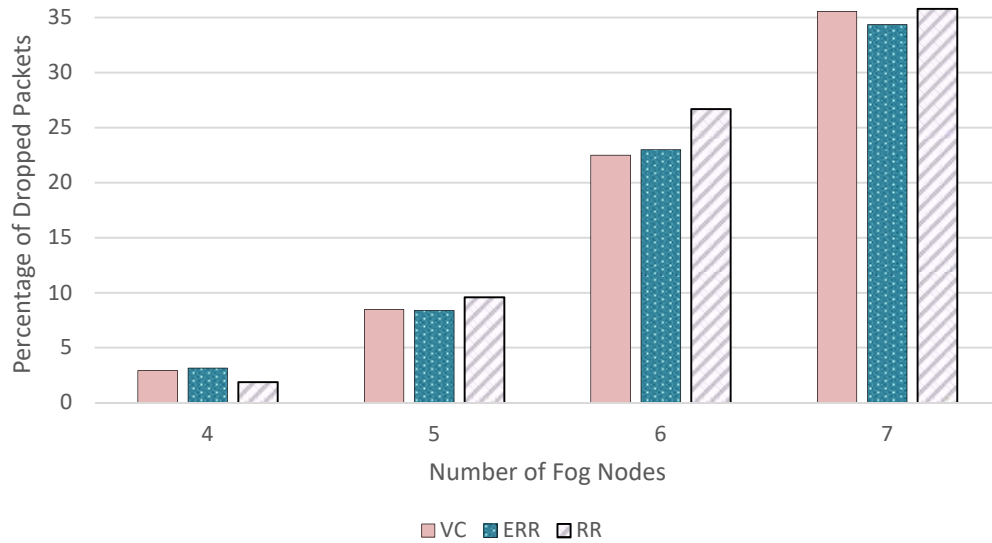
- But for classical round robin, fog nodes are only allowed to schedule their own packet requests and those coming from their RSUs. Increasing the number of fog nodes (which increases the number of needed slots), leads to the degradation of the performance of this access strategy, with nothing to minimize the effect of this problem. So this strategy degradation rate is greater than the other two strategies.



(a) Average Packet Queuing Delay



(b) Fairness Index



(c) Percentage of Dropped Packets

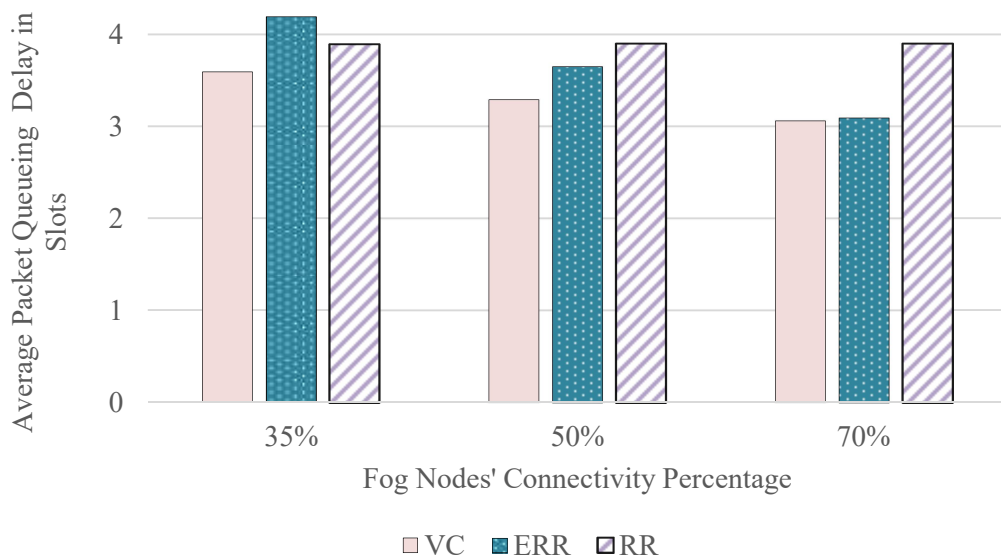
Figure 7.6 : Performance Evaluations of the three Access Strategies Using Different Number of Fog Nodes

### C. Changing network connectivity

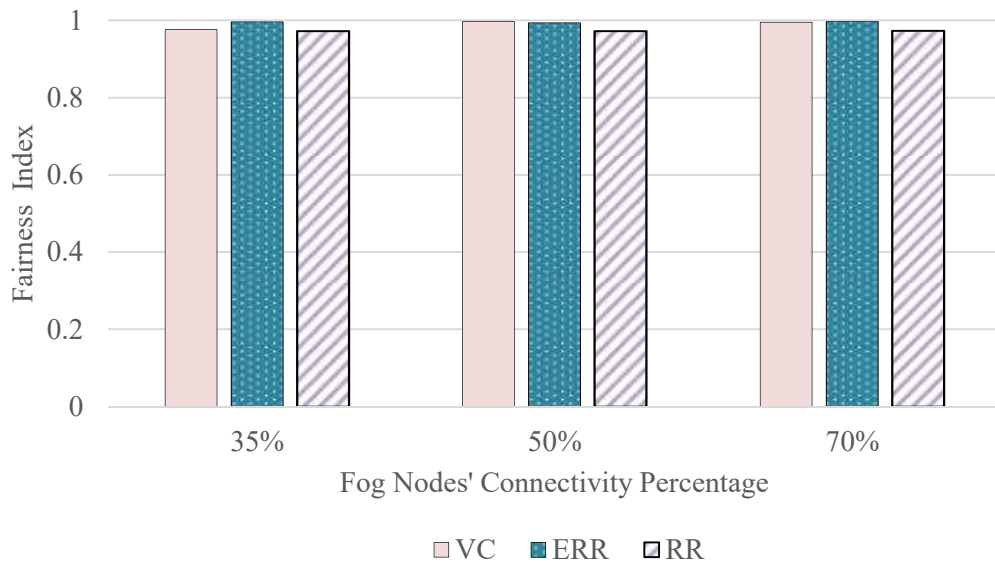
The graphs in Figure 7.7 illustrate the values of the three evaluated metrics resulted when applying the three access strategies together with the SDP-PSO protocol proposed in chapter 6, using various fog nodes network percentages connectivity, that vary from 35% to 70%. The number of RSUs per fog nodes and the number of fog nodes are kept constant, with values 8-10 and 6 respectively. The following points are the concluded driven from these graphs:

- Increasing the partial mesh network topology connectivity enhances all the three evaluated metric in both the vertex cover and the enhanced round robin access strategies. As increasing the connectivity will increase the number of each fog node neighbors, which will lead to increasing the fog nodes whose packets can be served by others, i.e. increases the number of fog nodes that are allowed to schedule their packets' request at each interval. So the packets' request may not have to stay long time in the queue before being sent, so both the average queuing delay and the percentage of dropped packets will decrease with the increase of the network connectivity.

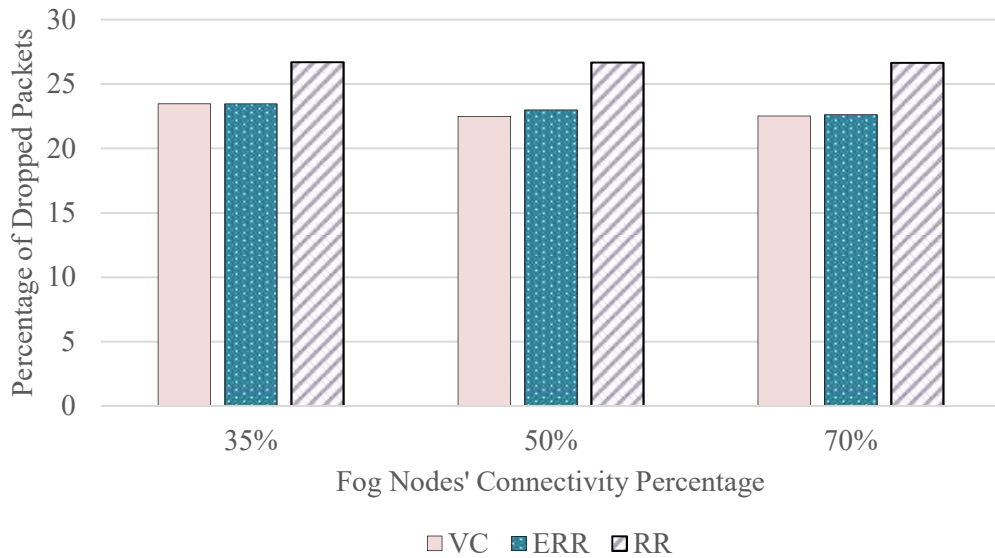
- Also the vertex cover performance is in a way better than the enhanced round robin as the cycle of vertex cover is shorter than that of the enhanced round robin, so the fog node's turn to schedule its packets comes faster in the vertex cover strategy than in the enhanced round robin.



(a) Average Packet Queuing Delay



(b) Fairness Index



(c) Percentage of Dropped Packets

Figure 7. 7 Performance Evaluations of the three Access Strategies Using Different Network Topology Connectivity

- However the network connectivity won't affect the performance of the classical round robin as the fog node is only allowed to schedule the packets' request formed by itself or those sent by it RSUs, they are not concerned with the number of its neighboring fog nodes.

### 7.3.4 Result Analysis

Table 7.3 summarizes the results of using the three access strategies and illustrates the percentage of improvement when using the Vertex Cover (VC) and Enhanced Round Robin (ERR) access strategies compared to the classical Round Robin access strategy.

The three access strategies; the vertex cover and the enhanced round robin and the classical round robin all performances are good, and can be applied but this depends on the network parameters. All have advantages and disadvantages. But in general, the classical round robin is better while only using a small number of fog nodes. The vertex cover and the enhanced round robin access strategies perform in general better than the classical round robin. As in those strategies the node in turn can schedule the packets' request formed by itself, those sent by their RSUs, and those sent by its neighboring fog nodes. This can be considered as the vertex cover

Table 7.3 Percentage of Improvement using the Proposed Vertex Cover and Enhanced Round Robin Access Strategies Compared to the Classical Round Robin Access Strategy

No of RSUs	Connectivity Percentage	No of fog Nodes	Changed Parameter	Performance Metric	Vertex Cover	Enhanced Round Robin	Round Robin	Average Percentage of improvement of Vertex Cover over Round Robin	Average Percentage of improvement of Enhanced Round Robin over Round Robin
Average Fairness Index	0.984	0.997	0.981	0.38%	1.68%				
Average Percentage of Dropped Packets	17.352	17.231	18.477	6.09%	6.74%				
Average Packet Queuing Delay in Slots	3.313	3.647	3.899	15.02%	6.47%				
Average Fairness Index	0.989	0.996	0.973	1.74%	2.37%				
Average Percentage of Dropped Packets	22.823	23.034	26.676	14.44%	13.65%				
Average Packet Queuing Delay in Slots	2.999	3.309	3.768	20.40%	12.19%				
Average Fairness Index	0.999	0.998	0.988	1.09%	0.99%				
Average Percentage of Dropped Packets	8.620	9.059	12.140	29.00%	25.38%				



and the enhanced round robin main advantage, which increase the chance of scheduling the packets' request in earlier time, i.e. each RSU may send multiple times in the same cycle. Results also show that vertex cover performs better than the enhanced round robin due to its short cycle, which decreases the time between two successive turns for each fog node.

The main disadvantage of the vertex cover access strategy is the discovery of the minimum vertex cover nodes at the set up phase. It is considered to be an NP-complete problem, however it is not a problem in small number of nodes, which is the case here. Note that all the fog nodes compete on the same channels as they are in the same area (channel reuse are not applied).

Also learning the network topology to establishing the cycle is an issue when dealing with the classical and the enhanced round robin access strategies. However another access strategy can be used in the set up phase e.g. random or pseudo random schemes to learn the topology. This is considered to be the main disadvantage in those access strategies.

## **7.4 Conclusion**

Applying the IoT smart traffic application; this chapter addressed the problem of distributed packets' scheduling for CRIoTs. It proposes using three distributed access strategies along with the SDP-PSO proposed in chapter 6 to schedule the packets on the estimated free slots resulted from applying the protocol proposed in chapter 5. The classical round robin and the two proposed strategies, the vertex cover and the enhanced round robin are the examined three access strategies.

An objective fitness function is used; it is near similar to the one used for the centralized protocol proposed in chapter 6 but with some differences according to its application for distributed scheduling. This objective fitness function combines three metrics; the fairness among RSUs, the packets' queuing delay and the number of dropped packets at the fog node responsible for the scheduling process at the time interval. The role of each access strategy is to specify the order by which the fog node will solve the scheduling problem as an optimization one using the proposed objective function by applying SDP-PSO to schedule the packets' request. It also specifies the fog nodes allowed to send packets at each time interval.

Results showed that classical round robin is better to be used in case of using small number of nodes. While in all other cases, vertex cover is the best one to be used. Enhanced round robin has an average performance in between the two other strategies.

# **Chapter 8**

## **Conclusions and Future Works**

# Chapter 8

## Conclusions and Future Works

This chapter summarizes the work in the thesis, it illustrates the conclusions obtained all over the thesis, and suggests some future works. It is divided into two sections; the first section overviewed the work and the obtained key conclusions. The second section points out to some of the possible future works.

### 8.1 Conclusion

In this thesis we proposed a complete system for spectrum allocation for CRIoT. So, our contribution is composed of three parts; the first one proposed two new protocols for channel availability prediction for interweave Cognitive Radio Networks (CRNs), both of them use Hidden Markov Model (HMM) in an advanced way. The proposed system consists of two stages, the training stage and the prediction stage. The training stage is common in the two proposed protocols; in which two HMM models (2-model HMM) are produced; the idle model and the busy model, they are trained each with its own sensing matrix training set. The sensing matrix is divided into two sets according to the last column in the matrix. The first set named the idle set contains the observations with rows ending by an idle state, and the second set named the busy set contains the rows ending with busy state. The idle set will be used to train the idle HMM model, while the busy set will be used to train the busy HMM model. In the prediction stage; the first protocol uses Bayes theorem and the 2-model HMM, while the second protocol uses Support Vector Machine (SVM) that employs the parameters produced from applying the 2-model HMM.

For performance evaluation, simulations of both proposed protocols are done together with prediction using classical HMM proposed. Three performance parameters were measured; the true percentage, the inaccuracy and the probability of PUs collision (false negative prediction). For the three protocols, those parameters are measured while changing the number of hidden states, the matrix's window size and the number of training days. Results showed that the proposed protocols perform better than the classical HMM under several conditions. Also, the 2-model HMM-

SVM outperforms the classical HMM and 2-model HMM, as it is trained with the HMM parameters, which are already trained. Also the feature vector produced for each sequence gives more precise information about the sensing slots in the sensing matrix.

In the second contribution, a centralized time slotted algorithm was proposed; it uses Discrete Permutation Particle Swarm Optimization (DP-PSO) for scheduling the packets among the free slots, obtained by applying the cognitive radio networks' channel estimation technique proposed in the first contribution. Applying smart healthcare facility as the deployed IoT application; the system configured in this phase consists of three main building blocks; the IoT devices block, the first layer fog nodes block and the central fog server. Each group of IoT devices is connected to a fog node; where they send data and receive orders. All fog nodes are connected to the central fog node. The protocol is named Scheduling based-on DP-PSO (SDP-PSO).

An objective function is used with the aim of maximizing the fairness index among the fog nodes, minimizing both the packets' queuing delay and the number of dropped unsend packets that exceeded their allowed time in the network. In order to evaluate the proposed SDP-PSO, it is simulated along with an old technique named spectrum auction protocol. Simulation results proved that the performance of the proposed SDP-PSO protocol overcomes that of the spectrum auction under different circumstances. It leads to better results in terms of the fairness, the average queuing delay and the number of dropped packets, while varying the number of IoT devices per fog node, the number of channels and the packets' rate. It was also proved that the SDP-PSO protocol has lower time and space complexity than the spectrum auction technique. SDP-PSO can be used in real time in contrary to the spectrum auction.

Finally the third part addressed the problem of distributed packets' scheduling for CRIoTs. Applying smart traffic as the deployed IoT application; the configured system in this part consists of three main building blocks; the IoT devices block, the first fog layer block (Road Side Units (RSUs)) and the second fog layer block. Each group of IoT devices is connected to a RSU, where they send data and receive orders. A group of RSUs are connected to a fog node which acts as their cluster head. This contribution proposes using three distributed access strategies along with the SDP-PSO proposed in the second part to schedule the packets on the estimated free slots, resulted from the applying the protocol proposed in the first contribution. The

examined three access strategies are the classical round robin strategy and two proposed strategies, the vertex cover and the enhanced round robin.

An objective fitness function is derived, it is near similar to the one used for the centralized protocol proposed in the second part, but with some differences to satisfy its application to distributed scheduling. This objective fitness function combines three metrics; the fairness among Road Side Units (RSUs), the packets' queuing delay and the number of dropped packets at the fog node responsible for the scheduling process at the time interval. The fairness index among RSUs, the average packet queuing delay and the percentage of dropped packets are the three performance metrics used for the evaluation of the proposed distributed packets' scheduling strategies for CRIoTs. Results proved that the classical round robin is better only while using a small number of fog nodes. The vertex cover and the enhanced round robin access strategies perform in general better than the classical round robin, as in those strategies the node in turn can schedule the packets request formed by itself, those sent by their RSUs, and those sent by its neighboring fog nodes. This can be considered as the vertex cover and the enhanced round robin main advantage, as it increases the chance of scheduling packet requests in earlier time, i.e. each RSU may send multiple times in the same cycle. Simulation results also prove that vertex cover performs better than the enhanced round robin due to its short cycle, which decreases the time between two successive turns for each fog node.

## 8.2 Future Works

Spectrum scarcity is an obstacle in IoT, so merging IoT and CRNs became a must for the IoT development to be able to grow as expected. Developing a system with this integration needs intensive research in several research points. So when speaking about the future works; many can be proposed; some of these are:

- Formulating a comprehensive IoT system architecture for developing all the CRNs functions; spectrum sensing, spectrum sharing, spectrum management and spectrum mobility.
- Integrating the 5G networks with CRIoTs for increasing the devices connectivity.
- Applying other more time sensitive IoT applications.

- Developing cloud fog platform data analysis architecture.
- Providing different levels of security for the hierarchical architecture, to protect the communications between layers.
- Deploying mobility for the various nodes, which will change the network topology.
- Using underlay Cognitive Radio Networks MAC protocols, which provide simultaneous communication with primary users, so it is more complex.
- Proposing distributed networks management, by allowing nodes to negotiate with each other to be self-organized and share the network resources effectively.
- Providing several heterogeneous devices, standards, technologies and protocols, for the coexistence of heterogeneous devices and protocols that need to operate with each other.
- Applying big data management by providing data classification, clustering, patterning and feature extraction for the huge data gathered for several applications.
- Providing scalable system by developing extendable services and operations to tolerate new services and devices joining the network constantly.

# References



## References

- [1] I. F. Akyildiz, W. Y. Lee, M. C. Vuran, and S. Mohanty, “NeXt Generation/Dynamic Spectrum Access/Cognitive Radio Wireless Networks: A survey,” *Computer Networks*, vol. 50, no. 13, pp. 2127–2159, 2006, doi: 10.1016/j.comnet.2006.05.001.
- [2] J. Tang, S. Misra, and G. Xue, “Joint Spectrum Allocation and Scheduling for Fair Spectrum Sharing in Cognitive Radio Wireless Networks,” *Computer Networks*, vol. 52, no. 11, pp. 2148–2158, 2008, doi: 10.1016/j.comnet.2008.03.010.
- [3] M. Renfors, “Cognitive Radio and its RF Challenges,” 2009. <http://www.cs.tut.fi/kurssit/ELT-44007/CR2012b.pdf>.
- [4] I. F. Akyildiz, W.-Y. Lee, M. Vuran, and M. Shantidev, “A Survey on Spectrum Management in Cognitive Radio Networks,” *IEEE Communications Magazine*, vol. 46, no. 4, pp. 40–48, 2008.
- [5] Anita Garhwal and P. Pratim Bhattacharya, “A survey on dynamic spectrum access techniques in cognitive radio networks,” *International Journal of Communication Networks and Information Security*, vol. 5, no. 2, pp. 68–79, 2011, doi: 10.5815/ijenis.2013.12.09.
- [6] V. Rohokale and N. Kulkarni, “Cooperative Opportunistic Large Array Approach for Cognitive Radio Networks,” in *Proceedings of 2010 8th International Conference on Communications, COMM 2010*, 2010, pp. 513–516, doi: 10.1109/ICCOMM.2010.5509024.
- [7] L. Chouhan and A. Trivedi, “Cognitive Radio Networks: Implementation and Application Issues in India,” 2011, [Online]. Available: <https://www.researchgate.net/publication/266677162>.
- [8] J. M. Iii, “Cognitive Radio An Integrated Agent Architecture for Software Defined Radio,” Royal Institute of Technology (KTH), Sweden., 2000.
- [9] H. Harada *et al.*, “A Software Defined Cognitive Radio System Cognitive Wireless Clouds,” *Proceeding of the IEEE Global Telecommunications*

- Conference (IEEE-GLOBECOM)*. pp. 294–299, 2007, doi: 10.1109/GLOCOM.2007.62.
- [10] B. Le, “Building a Cognitive Radio: From Architecture Definition to Prototype Implementation,” Faculty of the Virginia Polytechnic Institute and State University, 2007.
- [11] S. Li, J. Moskal, M. M. Kokar, and D. Brady, “An implementation of collaborative adaptation of cognitive radio parameters using an ontology and policy based approach,” *Analog Integrated Circuits and Signal Processing*, vol. 69, no. 2–3, pp. 283–296, 2011, doi: 10.1007/s10470-011-9681-y.
- [12] K. C. Chen, Y. J. Peng, N. Prasad, Y. C. Liang, and S. Sun, “Cognitive radio network architecture: Part I - General structure,” in *Proceedings of the 2nd International Conference on Ubiquitous Information Management and Communication, ICUIMC*, 2008, pp. 114–119, doi: 10.1145/1352793.1352817.
- [13] Y. Liang, L. Lai, and J. Halloran, “Distributed algorithm for collaborative detection in cognitive radio networks,” in *Proceeding of the 47th Annual Allerton Conference on Communication, Control, and Computing, Allerton*, 2009, pp. 394–399, doi: 10.1109/ALLERTON.2009.5394792.
- [14] M. Khasawneh and A. Agarwal, “A Survey on the Communication Protocols and Security in Cognitive Radio Networks,” in *Proceedings of the 6th International Conference on Computer Science and Information Technology, CSIT*, 2014, pp. 64–70, doi: 10.1109/CSIT.2014.6805980.
- [15] D. S. Vernekar, “An Investigation of Security Challenges in Cognitive Radio,” Faculty of the Graduate Coolege at the University of Nebraska, 2012.
- [16] R. Abhiraami and J. Kayalvizki, “A Perspective of Cognitive Radio in Wireless Sensor Networks – A Survey,” *International Journal of Advanced Research in Computer Science and Software Engineering (IJARCSSE)*, vol. 3, no. 1, pp. 354–357, 2013.
- [17] S. Parvin, F. K. Hussain, O. K. Hussain, S. Han, B. Tian, and E. Chang, “Cognitive radio network security: A survey,” *Journal of Network and*

- Computer Applications*, vol. 35, no. 6, pp. 1691–1708, 2012, doi: 10.1016/j.jnca.2012.06.006.
- [18] R. Ahmed and Y. A. Ghous, “Detection of Vacant Frequency Bands in Cognitive Radio,” Blekinge Institute of Technology, School of Engineering, 2010.
- [19] P. P. Bhattacharya, “A Novel Opportunistic Spectrum Access for Applications in Cognitive Radio,” *Ubiquitous Computing and Communication*, vol. 4, no. 2, pp. 24–28, 2011.
- [20] C. Chen, “Investigation of Primary User Emulation Attack in Cognitive Radio Networks,” Faculty of the Stevens Institute of Technology, Hoboken, NJ, 2010.
- [21] P. P. Bhattacharya, R. Khandelwal, R. Gera, and A. Agarwal, “Smart Radio Spectrum Management for Cognitive Radio,” *International Journal of Distributed and Paralle Systems (IJDPS)*, vol. 2, no. 4, pp. 12–24, 2011, doi: 10.5121/ijdps.2011.2402.
- [22] N. Yadav and S. Rathi, “Spectrum Sensing Techniques: Research, Challenge and Limitations 1 1,2,” *International Journal of Electronics & Communication Technology*, vol. 2, no. 4, pp. 240–245, 2011.
- [23] R. Rao, Q. Cheng, A. Kelkar, and D. Chaudhari, “Cooperative Cognitive Radio Network Testbed,” *ICST’s Global Community Magazine, ICaST*, 2011.
- [24] A. Naeem Akhtar and A. Masood Siddique, “Spectrum Decision Framework to Support Cognitive Radio Based IoT in 5G,” in *Cognitive Radio in 4G/5G Wireless Communication Systems Book*, 2018.
- [25] A. M. Rateb, “Introduction to Cognitive Radio Systems,” Universiti Teknologi Malaysia, 2008.
- [26] I. F. Akyildiz, W. Y. Lee, and K. R. Chowdhury, “CRAHNs: Cognitive radio ad hoc networks,” *Ad Hoc Networks*, vol. 7, no. 5, pp. 810–836, 2009, doi: 10.1016/j.adhoc.2009.01.001.
- [27] A. M. M. Almasoud, “Robust Provisioning of Multicast Sessions in Cognitive

- Radio Networks,” in *Proceeding of the 10th International Wireless Communications and Mobile Computing Conference (IWCMC)*, 2014, pp. 417–422, doi: 10.1109/IWCMC.2014.6906393.
- [28] J. Zhao and J. Yuan, “An Improved Centralized Cognitive Radio Network Spectrum Allocation Algorithm Based on the Allocation Sequence,” *International Journal of Distributed Sensor Networks*, vol. 9, no. 10, 2013, doi: 10.1155/2013/875342.
- [29] C. Cormio and K. R. Chowdhury, “A Survey on MAC Protocols for Cognitive Radio Networks,” *Ad Hoc Networks*, vol. 7, no. 7, pp. 1315–1329, 2009, doi: 10.1016/j.adhoc.2009.01.002.
- [30] N. Nguyen-Thanh, A. T. Pham, and Nguyen-Tam, “Medium Access Control Design for Cognitive Radio Networks: A Survey,” *IEICE Transactions on Communications*, vol. E97-B, no. 2, pp. 359–374, 2014, doi: 10.1587/transcom.E0.B.1.
- [31] T. Vamsi Krishna and A. Das, “A Survey on MAC Protocols in OSA Networks,” *Computer Networks*, vol. 53, no. 9, pp. 1377–1394, 2009, doi: 10.1016/j.comnet.2009.01.003.
- [32] V. U. Kanth, K. R. Chandra, and R. R. Kumar, “Spectrum Sharing in Cognitive Radio Networks,” *International Journal of Engineering Trends and Technology (IJETT)*, vol. 4, no. 4, pp. 1172–1175, 2013, doi: 10.1109/INFOCOM.2007.252.
- [33] B. Benmammar, A. Amraoui, and F. Krief, “A Survey on Dynamic Spectrum Access Techniques in Cognitive Radio Networks,” *International Journal of Communication Networks and Information Security (IJCNIS)*, vol. 5, no. 2, pp. 68–79, 2013, doi: 10.5815/ijcnis.2013.12.09.
- [34] S. Kandeepan, G. Baldini, and R. Piesiewicz, “UWB Cognitive Radios,” in *Novel Applications of the UWB Technologies Book*, 2011.
- [35] V. Katarwar and A. Kulkarni, “A Survey on Spectrum Management Framework in Cognitive Radio Networks—Research Challenges and Future Goals,” in *World Journal of Science and Technology*, 2012, vol. 2, no. 3, pp.

144–150.

- [36] A. Singh, “Study on Cognitive Radio and Cognitive Radio Network,” *International Journal of Emerging trends in Engineering and Development*, vol. 5, no. 2, pp. 238–245, 2012.
- [37] J. H. Martin, L. S. Dooley, and K. C. P. Wong, “A New Cross-Layer Design Strategy for TV White Space Cognitive Radio Applications,” in *Proceeding of the 3rd International Workshop on Cross Layer Design (IWCLD)*, 2011.
- [38] L. Lu, X. Zhou, U. Onunkwo, and G. Y. Li, “Ten Years of Research in Spectrum Sensing and Sharing in Cognitive Radio,” *EURASIP Journal on Wireless Communications and Networking*, vol. 2012, no. 28, pp. 1–16, 2012, doi: 10.1186/1687-1499-2012-28.
- [39] A. Khattab, D. Perkins, and M. Bayoumi, “Cognitive Radio Networking Preliminaries,” in *Cognitive Radio Networks: From Theory to Practice Book*, 2013.
- [40] R. Ferrús, O. Sallent, G. Baldini, and L. Goratti, “Public Safety Communications: Enhancement Through Cognitive Radio and Spectrum Sharing Principles,” *IEEE Vehicular Technology Magazine*, vol. 7, no. 2, pp. 54–61, 2012.
- [41] G. P. Villardi, G. T. F. de Abreu, and H. Harada, “TV White Space Technology: Interference in Portable Cognitive Emergency Network,” *IEEE Vehicular Technology Magazine*, vol. 7, no. 2, pp. 47–53, 2012, doi: 10.1109/MVT.2012.2190221.
- [42] L. Berlemann, S. Mangold, and A. Jarosch, “Operator Assisted Cognitive Radio for Dynamic Spectrum Access and Spectrum Sharing,” *Comsware*, vol. 1, pp. 1–6, 2006.
- [43] A. A. Khan, M. H. Rehmani, and A. Rachedi, “When Cognitive Radio meets the Internet of Things?,” *Proceeding of the International Wireless Communications and Mobile Computing Conference (IWCMC)*, pp. 469–474, 2016, doi: 10.1109/IWCMC.2016.7577103.

- [44] A. A. Khan, M. H. Rehmani, and A. Rached, "Cognitive-Radio-Based Internet of Things: Applications, Architectures, Spectrum Related Functionalities, and Future Research Directions," *IEEE Wireless Communications*, vol. 24, no. 3, pp. 17–25, 2017, doi: 10.1109/MCOM.2016.7514162.
- [45] P. Matta and B. Pant, "Internet-of-Things: Genesis, Challenges and Applications," *Journal of Engineering Science and Technology*, vol. 14, no. 3, pp. 1717–1750, 2019.
- [46] S. Kim, "Basic Concepts of Internet of Things and Game Theory," in *Securing the Internet of Things Book*, 2020, pp. 61–69.
- [47] Mimos Berhad, "National Internet of Things (IoT) Strategic Roadmap: A Summary," *MIMOS Berhad*, 2015. [http://www.mimos.my/iot/National\\_IoT\\_Strategic\\_Roadmap\\_Summary.pdf](http://www.mimos.my/iot/National_IoT_Strategic_Roadmap_Summary.pdf).
- [48] M. Muntjir, M. Rahul, and H. A. Alhumyani, "An Analysis of Internet of Things (IoT): Novel Architectures, Modern Applications, Security Aspects and Future Scope with Latest Case Studies," *International Journal of Engineering Research & Technology (IJERT)*, vol. 6, no. 06, pp. 422–448, 2017.
- [49] O. Vermesan *et al.*, "Internet of Things Strategic Research Roadmap," *Internet of Things: Global Technological and Societal Trends*, vol. 1, pp. 9–52, 2011.
- [50] M. Burhan, R. A. Rehman, B. Khan, and B. S. Kim, "IoT Elements, Layered Architectures and Security Issues: A Comprehensive Survey," *Sensors Journal*, vol. 18, no. 9, 2018, doi: 10.3390/s18092796.
- [51] P. Sethi and S. R. Sarangi, "Internet of Things: Architectures, Protocols, and Applications," *Journal of Electrical and Computer Engineering*, vol. 2017, no. 1, pp. 1–25, 2017, doi: 10.1155/2017/9324035.
- [52] S. Balaji, K. Nathani, and R. Santhakumar, "IoT Technology, Applications and Challenges: A Contemporary Survey," *Wireless Personal Communications*, vol. 108, no. 1, pp. 363–388, 2019, doi: 10.1007/s11277-019-06407-w.

- [53] A. El Hakim, “Internet of Things (IoT) System Architecture and Technologies,” *White Paper*, 2018. [https://www.researchgate.net/profile/Ahmed\\_El\\_Hakim2/publication/323525875\\_Internet\\_of\\_Things\\_IoT\\_System\\_Architecture\\_and\\_Technologies\\_White\\_Paper/links/5a99c15aaca2721e3f2de808/Internet-of-Things-IoT-System-Architecture-and-Technologies-White-Paper.pdf](https://www.researchgate.net/profile/Ahmed_El_Hakim2/publication/323525875_Internet_of_Things_IoT_System_Architecture_and_Technologies_White_Paper/links/5a99c15aaca2721e3f2de808/Internet-of-Things-IoT-System-Architecture-and-Technologies-White-Paper.pdf).
- [54] O. Elijah, T. A. Rahman, I. Orikumhi, C. Y. Leow, and M. N. Hindia, “An Overview of Internet of Things (IoT) and Data Analytics in Agriculture: Benefits and Challenges,” *IEEE Internet of Things Journal*, vol. 5, no. 5, pp. 3758–3773, Oct. 2018, doi: 10.1109/JIOT.2018.2844296.
- [55] Sanchit, “The Internet of Things-Introduction and Applications,” *International Research Journal of Engineering and Technology (IRJET)*, vol. 5, no. 8, pp. 2255–2262, 2018, doi: 10.1155/2017/9324035.
- [56] N. S. Swamy, S. Nayak, and V. M. N, “Analysis on IoT Challenges, Opportunities, Applications and Communication Models,” *International Journal of Advanced Engineering, Management and Science (IJAEMS)*, vol. 2, no. 4, pp. 75–78, 2016.
- [57] P. Parhana, M. Lakshmaiah, S. Allaudheen, S. Dastagiri, and M. V. Saritha, “Review on Internet of Things: Recent Applications and its Challenges,” *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, vol. 6, no. 11, 2017.
- [58] G. Marques, N. Garcia, and N. Pombo, “A Survey on IoT: Architectures, Elements, Applications, QoS, Platforms and Security Concepts,” in *Advances in Mobile Cloud Computing and Big Data in the 5G Era Book*, C. X. Mavromoustakis, G. Mastorakis, and Ciprian Dobre, Eds. 2017, pp. 115–130.
- [59] G. S. Iyer, “Internet of Things – A Dynamic Global Network Infrastructure,” *International Journal of Innovative Reserach in Science Engineering and Technology*, vol. 7, no. 1, pp. 663–669, 2018, doi: 10.15680/IJRSET.2018.0701096.
- [60] P. Masek *et al.*, “A harmonized perspective on transportation management in

smart cities: The novel IoT-driven environment for road traffic modeling,” *Sensors (Switzerland)*, vol. 16, no. 11, 2016, doi: 10.3390/s16111872.

- [61] H. Shahinzadeh, J. Moradi, G. B. Gharehpetian, H. Nafisi, and M. Abedi, “IoT Architecture for smart grids,” *International Conference on Protection and Automation of Power System, IPAPS 2019*, no. March, pp. 22–30, 2019, doi: 10.1109/IPAPS.2019.8641944.
- [62] Y. Abuseta, “A Fog Computing Based Architecture for IoT Services and Applications Development,” *International Journal of Computer Trends and Technology*, vol. 67, no. 10, pp. 92–98, 2019, doi: 10.14445/22312803/ijctt-v67i10p116.
- [63] M. Yannuzzi, R. Milito, R. Serral-Gracia, D. Montero, and M. Nemirovsky, “Key ingredients in an IoT recipe: Fog Computing, Cloud computing, and more Fog Computing,” in *IEEE 19th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks, CAMAD*, 2014, pp. 325–329, doi: 10.1109/CAMAD.2014.7033259.
- [64] C. C. Byers and P. Wetterwald, “Fog Computing Distributing Data and Intelligence for Resiliency and Scale Necessary for IoT,” in *The Internet of Things Ubiquity Symposium*, 2015, pp. 1–12, doi: 10.1145/2822875.
- [65] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, “Fog Computing and Its Role in the Internet of Things,” in *Proceeding of the First Edition of the MCC Workshop on Mobile Cloud Computing*, 2012, pp. 13–16, doi: 10.1177/1464419316662568.
- [66] F. A. Awin, Y. M. Alginahi, E. Abdel-Raheem, and K. Tepe, “Technical Issues on Cognitive Radio-Based Internet of Things Systems: A Survey,” *IEEE Access*, vol. 7, pp. 97887–97908, 2019, doi: 10.1109/ACCESS.2019.2929915.
- [67] Q. Wu *et al.*, “Cognitive internet of things: A new paradigm beyond connection,” *IEEE Internet of Things Journal*, vol. 1, no. 2, pp. 129–143, 2014, doi: 10.1109/JIOT.2014.2311513.
- [68] K. Katzis and H. Ahmadi, “Challenges Implementing Internet of Things (IoT)



Using Cognitive Radio Capabilities in 5G Mobile Networks,” in *Internet of Things (IoT) in 5G Technologies Book*, vol. 8, 2016, pp. 55–76.

- [69] F. F. Qureshi, R. Iqbal, and M. N. Asghar, “Energy efficient wireless communication technique based on Cognitive Radio for Internet of Things,” *Journal of Network and Computer Applications*, vol. 89, pp. 14–25, 2017, doi: 10.1016/j.jnca.2017.01.003.
- [70] B. Zhang and K. Chen, “Selective spectrum leasing in Internet of Things via Nash Bargaining Solutions,” in *Proceedings of the IEEE 2nd International Conference on Cloud Computing and Intelligence Systems, (IEEE CCIS)*, 2012, vol. 2, pp. 841–845, doi: 10.1109/CCIS.2012.6664294.
- [71] B. Moon, “Dynamic spectrum access for internet of things service in cognitive radio-enabled LPWANs,” *Sensors Journal*, vol. 17, no. 12, 2017, doi: 10.3390/s17122818.
- [72] H. Kim, S. W. Ko, and S.-L. Kim, “Cognitive Random Access for Internet-of-Things Networks,” in *Proceeding of the IEEE 85th Vehicular Technology Conference*, 2017.
- [73] J. Wen, Q. Yang, and S.-J. Yoo, “Optimization of Cognitive Radio Secondary Information Gathering Station Positioning and Operating Channel Selection for IoT Sensor Networks,” *Mobile Information Systems Journal*, vol. 2018, pp. 1–12, 2018, doi: 10.1155/2018/4721956.
- [74] L. Qu, Y. Huang, M. Shi, and C. Tang, “A Spectrum Allocation System Model of Internet of Things Based on Cognitive Radio,” in *Knowledge Discovery and Data Mining Book*, 2012, pp. 177–184.
- [75] R. F. Shigueta, M. Fonseca, A. Ziviani, A. C. Viana, and A. Munaretto, “A Strategy for Opportunistic Cognitive Channel Allocation in Wireless Internet of Things,” *Proceeding of the 2014 IFIP Wireless Days (WD) Conference*, 2014.
- [76] S. Kim, “Inspection game based cooperative spectrum sensing and sharing scheme for cognitive radio IoT system,” *Computer Communications Journal*, vol. 105, pp. 116–123, 2017, doi: 10.1016/j.comcom.2017.01.015.

- [77] X. Liu and J. Xie, "Priority-based spectrum access in cognitive D2D networks for IoT," *Proceeding of the IEEE International Conference on Communications (ICC)*, 2017, doi: 10.1109/ICC.2017.7996716.
- [78] S. Mohapatra, P. K. Sahoo, and J. P. Sheu, "Spectrum Allocation with Guaranteed Rendezvous in Asynchronous Cognitive Radio Networks for Internet of Things," *IEEE Internet of Things Journal*, vol. 6, no. 4, pp. 6104–6116, 2018, doi: 10.1109/JIOT.2018.2872459.
- [79] J. Zhu, Y. Song, D. Jiang, and H. Song, "A New Deep Q-Learning-Based Transmission Scheduling Mechanism for the Cognitive Internet of Things," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2375–2385, 2018, doi: 10.1109/JIOT.2017.2759728.
- [80] R. Han, Y. Gao, C. Wu, and D. Lu, "An Effective Multi-Objective Optimization Algorithm for Spectrum Allocations in the Cognitive-Radio-Based Internet of Things," *IEEE Access Journal*, vol. 6, pp. 12858–12867, 2018, doi: 10.1109/ACCESS.2017.2789198.
- [81] H. B. Salameh, S. Almasri, E. Benkhelifa, and J. Lloret, "Spectrum Assignment in Hardware-constrained Cognitive Radio IoT Networks under Varying Channel-quality Conditions," *IEEE Access Journal*, vol. 7, pp. 42816–42825, 2019, doi: 10.1109/access.2019.2901902.
- [82] P. P. Roy, "Hidden Markov Model based Channel state prediction in Cognitive Radio Networks," *International Journal of Engineering Research & Technology (IJERT)*, vol. 4, no. 2, pp. 391–394, 2015.
- [83] V. K. Tumuluru, P. Wang, and D. Niyato, "Channel Status Prediction for Cognitive Radio Networks," *Wireless Communications and Mobile Computing*, vol. 12, no. 10, pp. 862–874, 2010, doi: 10.1002/wcm.
- [84] S. Perumal, V.M. Gayathri, and R. Pitchai, "Optimized Neural Network for Spectrum Prediction Using Genetic Algorithm in Cognitive Radio Networks," *Cluster Computing Journal*, vol. 22, pp. 157–163, 2019.
- [85] X. Chen, Z. Chen, S. Xie, and Y. Shao, "Predictive and Recommendatory Spectrum Decision for Cognitive Radio," 2017.

<https://arxiv.org/pdf/1703.03102.pdf>.

- [86] A. Canavitsas, L. S. Mello, and M. Grivet, “Spectral Vacancies Prediction Method for Cognitive Radio Applications,” *Journal of Microwaves, Optoelectronics and Electromagnetic Applications*, vol. 15, no. 1, pp. 18–29, 2016.
- [87] S. Jain and A. Goel, “A Survey of Spectrum Prediction Techniques for Cognitive Radio Networks,” *International Journal of Applied Engineering Research*, vol. 12, no. 10, pp. 2196–2201, 2017.
- [88] L. M. Tuberquia-david, L. Cruz, and C. Hernández, “Spectral Prediction : Approaches in Cognitive Radio Networks,” *International Journal of Applied Engineering Research*, vol. 13, no. 10, pp. 8051–8063, 2018, doi: 10.1109/NGMAST.2014.40.
- [89] L. R. Rabiner and B. H. Juang, “An Introduction to Hidden Markov Models,” *IEEE ASSP Magazine*, vol. 3, no. 1, pp. 4–16, 1986, doi: 10.12835/ve2017.2-0095.
- [90] A. Pradhan, “Support Vector Machine-A Survey,” *International Journal of Emerging technology Technology and Advanced Engineering*, vol. 2, no. 8, pp. 82–85, 2012.
- [91] J. Nayak, B. Naik, and H. S. Behera, “A Comprehensive Survey on Support Vector Machine in Data Mining Tasks: Applications & Challenges,” *International Journal of Database Theory and Application*, vol. 8, no. 1, pp. 169–186, 2015, doi: 10.14257/ijdta.2015.8.1.18.
- [92] Java T Point, “Support Vector Machine Algorithm,” 2018. <https://www.javatpoint.com/machine-learning-support-vector-machine-algorithm>.
- [93] MathWorks, “MATLAB,” *Www.Mathworks.Com*, 2013. .
- [94] L. Refaeilzadeh, Payam Tang and H. Liu, “Cross-Validation,” *Encyclopedia of Database Systems*. pp. 532–538, 2009, doi: 10.5743/cairo/9789774160097.003.0002.

- [95] M. Khaledi and A. A. Abouzeid, "Optimal Bidding in Repeated Wireless Spectrum Auctions with Budget Constraints," in *Proceeding of 2016 IEEE Global Communications Conference, GLOBECOM 2016*, 2016, pp. 1–20, doi: 10.1109/GLOCOM.2016.7842076.
- [96] S. D. Chavan and Nisha P. Adgokar, "An Overview on Particle Swarm Optimization: Basic Concepts and Modified Variants," *International Journal of Science and Research (IJSR)*, vol. 4, no. 5, pp. 255–260, 2015.
- [97] J. Blondin, "Particle Swarm Optimization: A Tutorial," 2009. <https://pdf4pro.com/view/particle-swarm-optimization-a-tutorial-5b8743.html>.
- [98] P. B. Mat, "Introduction to Particles Swarm Optimization," 2013. <https://pt.slideshare.net/raafiubrian/particles-swarm-optimization>.
- [99] N. Hossein, R. Majid, and M.-F. Malihe, "Binary Particle Swarm Optimization: challenges and New Solutions," *Journal of Computer Society of Iran (CSI) On Computer Science and Engineering (JCSE)*, vol. 6, no. (1-A), pp. 21–32, 2008.
- [100] S. Mirjalili and A. Lewis, "S-shaped versus V-shaped transfer functions for binary Particle Swarm Optimization," *Swarm and Evolutionary Computation*, vol. 9, pp. 1–14, 2013, doi: 10.1016/j.swevo.2012.09.002.
- [101] S. Yang, M. Wang, and L. Jiao, "A Quantum Particle Swarm Optimization," in *Proceeding of the 2004 Congress on Evolutionary Computation, (CEC2004)*, 2004, vol. 1, pp. 320–324.
- [102] Q. K. Pan, M. Fatih Tasgetiren, and Y. C. Liang, "A Discrete Particle Swarm Optimization Algorithm for the Permutation Flowshop Sequencing Problem with Makespan Criterion," *Research and Development in Intelligent Systems XXIII - Proceedings of AI 2006, the 26th SGA International Conference on Innovative Techniques and Applications of Artificial Intelligence*, pp. 19–31, 2007, doi: 10.1007/978-1-84628-663-6-2.
- [103] P. Bhardwaj *et al.*, "Enhanced Dynamic Spectrum Access in Multiband Cognitive Radio Networks via Optimized Resource Allocation," *IEEE Transactions on Wireless Communications*, vol. 15, no. 12, pp. 8093–8106,

2016, doi: 10.1109/TWC.2016.2612627.

- [104] B. S. Awoyemi, B. T. Bodhaswar, and A. S. Alfa, “Solving Resource Allocation Problems in Cognitive Radio Networks: a Survey,” *EURASIP Journal on Wireless Communications and Networking*, vol. 2016, no. 1, 2016, doi: 10.1186/s13638-016-0673-6.
- [105] R. K. Jain, D.-M. W. Chiu, and W. R. Hawe, “A Quantitative Measure of Fairness And Discrimination For Resource Allocation In Shared Computer Systems,” 1984. [Online]. Available: <http://arxiv.org/abs/cs/9809099>.
- [106] M. T. M. Emmerich and A. H. Deutz, “A Tutorial on Multiobjective Optimization: Fundamentals and Evolutionary Methods,” *Natural Computing Journal*, vol. 17, no. 3, pp. 585–609, 2018, doi: 10.1007/s11047-018-9685-y.
- [107] X.-S. Yang, “Multi-Objective Optimization,” *Nature-Inspired Optimization Algorithms*, pp. 197–211, 2014, doi: 10.1016/b978-0-12-416743-8.00014-2.
- [108] Steven S. Skeina, *The Algorithm Design Manual*, Second edi. Springer-Verlag London Limited, 2008.
- [109] R. Craciunescu, A. Mihovska, M. Mihaylov, S. Kyriazakos, R. Prasad, and S. Halunga, “Implementation of Fog Computing for Reliable E-Health Applications,” *Proceeding of the 49th Asilomar Conference on Signals, Systems and Computers*, pp. 459–463, 2015, doi: 10.1109/ACSSC.2015.7421170.
- [110] C. Habib, “Energy-efficient Data Collection and Fusion in Wireless Body Sensor Networks for Continuous Health Monitoring,” Universite Bourgogne France Comte (UBFC), 2018.
- [111] G. L. NeuroTechnologies, “Health Sensors Index.” <https://glneurotech.com/FAQ/>.
- [112] J. R. Parvin, “An Overview of Wireless Mesh Networks,” in *Wireless Mesh Networks Security, Architecture and Protocols Book*, 2019.
- [113] A. Cilfone, L. Davoli, L. Belli, and G. Ferrari, “Wireless Mesh Networking: An IoT-Oriented Perspective Survey on Relevant Technologies,” *Future*

*Internet Journal*, vol. 11, no. 4, 2019, doi: 10.3390/fi11040099.

- [114] K. S. Vijayalayan, A. Harwood, and S. Karunasekera, “Distributed Scheduling Schemes for Wireless Mesh Networks: A Survey,” *ACM Computing Surveys*, vol. 46, no. 1, 2013, doi: 10.1145/2522968.2522982.
- [115] W. Stallings, *Data and Computer Communications*, 10th ed. Prentice Hall, 2014.
- [116] M. Chen, Y. Tian, G. Fortino, J. Zhang, and I. Humar, “Cognitive Internet of Vehicles,” *Computer Communications Journal*, vol. 120, pp. 58–70, 2018, doi: 10.1016/j.comcom.2018.02.006.
- [117] O. Kaiwartya *et al.*, “Internet of Vehicles : Motivation , Layered Architecture , Network Model , Challenges , and Future Aspects,” *IEEE Access*, vol. 4, pp. 5356–5373, 2016, doi: 10.1109/ACCESS.2016.2603219.
- [118] J. Wang, C. Jiang, K. Zhang, T. Q. S. Quek, Y. Ren, and L. Hanzo, “Vehicular Sensing Networks in a Smart City: Principles, Technologies and Applications,” *IEEE Wireless Communications*, vol. 25, no. 1, pp. 122–132, 2018, doi: 10.1109/MWC.2017.1600275.
- [119] E. M. Tordera *et al.*, “What is a Fog Node A Tutorial on Current Concepts Towards a Common Definition,” Cornell University, 2016.
- [120] K. Dolui and S. K. Datta, “Comparison of Edge Computing Implementations: Fog Computing, Cloudlet and Mobile Edge Computing,” *Proceeding of 2017 - Global Internet of Things Summit (GIoTS)*. 2017, doi: 10.1109/GIOTS.2017.8016213.
- [121] G. Karakostas, “A Better Approximation Ratio for The Vertex Cover Problem,” *International Colloquium on Automata, Languages, and Programming*, vol. July, pp. 1043–1050, 2005, doi: 10.1145/1597036.1597045.
- [122] S. Balaji, V. Swaminathan, and K. Kannan, “Optimization of Unweighted Minimum Vertex Cover,” *International Journal of Mathematical and Computational Sciences.*, vol. 43, no. 7, pp. 716–729, 2010, doi:

10.5281/zenodo.1079664.

# **Outcome from the Thesis**



## Outcome from the Thesis

D. Tarek, A. Benslimane, M. Darwish, and A. M. Kotb, "Cognitive Radio Networks Channel State Estimation Using Machine Learning Techniques," in 2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC), 2019, pp. 342–347.

D. Tarek, A. Benslimane, M. Darwish, and A. M. Kotb, "Survey on spectrum sharing/allocation for cognitive radio networks Internet of Things," Egypt. Informatics Journal, 2020. doi:10.1016/j.eij.2020.02.003

D. Tarek, A. Benslimane, M. Darwish, and A. M. Kotb, "A new strategy for packets scheduling in cognitive radio internet of things," Computer Networks Journal, vol. 178, no. April, 2020. doi: 10.1016/j.comnet.2020.107292

D. Tarek, A. Benslimane, M. Darwish, and A. M. Kotb, "Proposed Distributed Packets Scheduling Technique for Cognitive Radio Internet of Things Based on Discrete Permutation Particle Swarm Optimization", in the 13th IEEE International Conference on Internet of Things, 2020