

AN OMNET++ MODEL FOR SU TRANSMISSIONS WITH PU PROTECTION IN A  
COGNITIVE RADIO NETWORK



by  
Beyza Gölge

Submitted to Graduate School of Natural and Applied Sciences  
in Partial Fulfillment of the Requirements  
for the Degree of Master of Science in  
Electrical and Electronics Engineering

Yeditepe University

2022

AN OMNET++ MODEL FOR SU TRANSMISSIONS WITH PU PROTECTION IN A  
COGNITIVE RADIO NETWORK

APPROVED BY:

Assoc. Prof. Dr. Engin Maşazade  
(Thesis Supervisor)  
(Marmara University)

.....

Assist. Prof. Dr. Uğur Yıldırım  
(Yeditepe University)

.....

Prof. Dr. Özgür Gürbüz Ünlüyurt  
(Sabanci University)

.....

DATE OF APPROVAL: ...../...../2022

I hereby declare that this thesis is my own work and that all information in this thesis has been obtained and presented in accordance with academic rules and ethical conduct. I have fully cited and referenced all material and results as required by these rules and conduct, and this thesis study does not contain any plagiarism. If any material used in the thesis requires copyright, the necessary permissions have been obtained. No material from this thesis has been used for the award of another degree.

I accept all kinds of legal liability that may arise in case contrary to these situations.

Name, Last name      Beyza Gölge

Signature

## **ACKNOWLEDGEMENTS**

I would like to thank my thesis advisor, Engin Maşazade, for giving me this opportunity. I am grateful for patiently guiding and supporting me during my thesis study. I want to thank my parents for their unconditional support and for their understanding my ups and downs during this process. Many thanks to my friends for always being there when I need it.



## ABSTRACT

### AN OMNET++ MODEL FOR SU TRANSMISSIONS WITH PU PROTECTION IN A COGNITIVE RADIO NETWORK

The studies in cognitive radio networks (CRN) mostly focus on how the secondary users (SU) can use the spectrum most efficiently if there is no primary user (PU) transmitting in the medium. However, PU transmission must be protected while SU utilizes medium successfully. For this reason, in this thesis, we take into account PU transmission and studies are done to minimize the destructive effects of SU transmissions on PU transmission. We develop two different interweave spectrum sharing CRN models in the OMNET++ simulation program. In our first study, we model a single-channel CRN with a PU and a SU transceiver pairs based on INET Framework. One of the two modules that INET offers as a Medium Access Control (MAC) protocol, the Carrier Sense Multiple Access / Collision Avoidance (CSMA/CA) based CsmCaMac module is suitable for SU. The other MAC protocol module offered, AckingMac, allows the user to transmit without listening to the medium, but it is not compatible with the CsmCaMac module and does not have enough features for PU transmission. Therefore, we have developed a new module for PU transmission, which is compatible with the CsmCaMac module. The PU module enables the user to transmit the medium without listening and has a retransmission and an acknowledgment (ACK) mechanism. Our second study is two-channel CRN model that we developed based on the crSimulator study. While crSimulator provides infrastructure for multi-channel CRN, it does not provide real communication for PUs. The PU node is only used as a transmitter to keep the medium busy. Packets transmitted from PU do not have a receiving node and point-to-point transmission is not performed. Therefore, in our two-channel CRN model, we model both receiver and transmitter PU nodes and guarantee the transmission of packets to the peer node with a retransmission and an ACK mechanism. We verify the two developed CRN models by taking various simulations. We examine the packet transmission for both users by changing packet sizes and the interval time of the PUs and performing the SU transmission over a relay node.

## ÖZET

### BİLİŞSEL RADYO AĞINDA PU KORUMALI SU İLETİMLERİ İÇİN OMNET++ MODELİ

Bilişsel radyo ağlarında (CRN) çalışmalar daha çok ikincil kullanıcıların (SU) spectrumu ortamda iletim yapan birincil kullanıcı (PU) yoksa nasıl en verimli şekilde kullanacağı üzerinedir. Ancak SU kendi iletimini başarılı bir şekilde yaparken PU başarımının korunması da gerekmektedir. Bu nedenle bu tezde PU iletime odaklanılmış, PU iletimlerinin SU iletiminden en az düzeyde etkilenmesine yönelik çalışmalar yapılmıştır. Bunun için iki farklı araya yerleştirmeli spectrum paylaşım CRN modelini OMNET++ simülasyon programında geliştiriyoruz. İlk çalışmamızda, bir PU ve bir SU alıcı verici çifti ile araya yerleştirmeli tek kanallı INET Framework'e dayalı CRN modelliyoruz. INET'in Ortam Erişim Kontrol (MAC) protokolü olarak sunduğu iki modülden biri olan Çarpışmadan Kaçınmayla Taşıyıcı Dinleyen Çoklu Erişim (CSMA/CA) tabanlı CsmacaMac modülü SU için uygundur. Sunulan diğer MAC protokol modülü AckingMac, kullanıcının ortamı dinlemeden iletim yapmasını sağlar ancak CsmacaMac modülü ile uyumlu değildir ve PU iletimi için yeterli özelliklere sahip değildir. Bu nedenle, PU iletimi için CsmacaMac modülü ile uyumlu yeni bir modül geliştirdik. PU modülü, kullanıcının ortamı dinlemeden ilemesini sağlar ve yeniden iletim ve alındı (ACK) mekanizmasına sahiptir. İkinci çalışmamız ise crSimulator çalışmasını baz alarak oluşturduğumuz iki kanallı CRN modelidir. crSimulator çoklu kanal CRN için alt yapı sunarken PU'lar için gerçek bir haberleşme sağlamaz. PU düğümü yalnızca ortamı meşgul tutmak için verici olarak kullanılır. PU paketlerinin alıcı düğümü yoktur noktadan noktaya iletim gerçekleşmez. Bu nedenle iki kanallı CRN modelimizde hem alıcı hem verici PU düğümleri modelleyip paketlerin karşı düğüme iletimini yeniden iletim ve ACK mekanizması ile garantiliyoruz. Geliştirilen iki CRN modelini çeşitli simülasyonlar alarak doğruluyoruz. Paket boyutlarını, PU'ların ortama çıkış sürelerini değiştirerek ve SU iletimini röle aracılığıyla gerçekleştirerek her iki kullanıcı için paket iletimini inceliyoruz.

## TABLE OF CONTENTS

ACKNOWLEDGEMENTS .....	iv
ABSTRACT.....	v
ÖZET .....	vi
LIST OF FIGURES .....	ix
LIST OF TABLES .....	xii
LIST OF SYMBOLS/ABBREVIATIONS.....	xiii
1. INTRODUCTION.....	15
1.1. LITERATURE SURVEY .....	18
1.2. LIST OF CONTRIBUTIONS .....	22
1.3. THESIS ORGANISATION .....	23
2. PRELIMINARIES.....	25
2.1. OMNET++ .....	25
2.2. MAC PROTOCOL IN IEEE 802.11.....	25
2.2.1. CSMA/CA Technique.....	26
2.2.2. RTS/CTS Technique.....	29
2.2.3. Backoff Algorithm.....	31
3. CRN MODEL UNDER SINGLE CHANNEL.....	33
3.1. INET FRAMEWORK.....	33
3.1.1. Architecture of Host.....	33
3.1.2. MAC Modules for Wireless Communication in INET.....	37
3.2. PU MODIFIED MAC LAYER.....	38
3.3. SU MAC LAYER .....	42
3.4. PHY LAYER .....	44
4. CRN MODEL UNDER TWO CHANNEL.....	47
4.1. PU HOST ARCHITECTURE.....	47
4.2. MODIFIED PU HOST ARCHITECTURE .....	47
4.2.1. APP Layer for Modified PU .....	48

4.2.2.	MAC Layer for Modified PU .....	49
4.3.	SU HOST ARCHITECTURE.....	50
4.3.1.	Layers of Protocol Stack.....	51
4.3.2.	MAC Layer of SU.....	51
4.3.3.	Modules for SU Host .....	53
4.3.3.1.	Spectrum Sensor Module .....	53
4.3.3.2.	RF Spectrum Module .....	54
4.3.4.	Modified Spectrum Sensor Module .....	54
4.3.5.	MAC Layer with Spectrum Sensor.....	55
4.4.	CRN MODEL IN CRSIMULATOR .....	57
4.5.	DEVELOPED CRN MODEL WITH TWO CHANNEL .....	58
5.	RESULTS AND DISCUSSIONS .....	59
5.1.	CRN MODEL UNDER SINGLE CHANNEL RESULTS.....	59
5.1.1.	Simulation Parameters .....	60
5.1.2.	Simulation Scenarios .....	61
5.1.2.1.	First Scenario.....	61
5.1.2.2.	First Scenario Results .....	62
5.1.2.3.	Second Scenario .....	65
5.1.2.4.	Second Scenario Results.....	66
5.2.	CRN MODEL UNDER TWO CHANNEL RESULTS.....	69
5.2.1.	Simulation Parameters .....	69
5.2.2.	Simulation Scenarios .....	70
5.2.2.1.	First Scenario.....	70
5.2.2.2.	First Scenario Results .....	71
5.2.2.3.	Second Scenario .....	74
5.2.2.4.	Second Scenario Results.....	75
5.2.2.5.	Third Scenario .....	78
5.2.2.6.	Third Scenario Results .....	80
6.	CONCLUSION AND FUTURE WORK.....	84
	REFERENCES .....	87

## LIST OF FIGURES

Figure 1.1. Spectrum utilization .....	15
Figure 1.2. Spectrum hole (White space) .....	16
Figure 1.3. Cognitive cycle.....	18
Figure 2.1. CSMA/CA working mechanism .....	28
Figure 2.2. Flowchart of CSMA/CA at transmitter node .....	28
Figure 2.3. Flowchart of RTS/CTS at transmitter node.....	30
Figure 2.4. RTS/CTS and backoff mechanism of DCF.....	30
Figure 2.5. Backoff algorithm in IEEE 802.11 .....	32
Figure 3.1. Host module in CRN .....	33
Figure 3.2. APP module in CRN .....	34
Figure 3.3. UDP Sink module in CRN .....	34
Figure 3.4. Transport layer module in CRN .....	34
Figure 3.5. Network layer module in CRN.....	34
Figure 3.6. Radio / MAC module .....	35
Figure 3.7. MAC queue module .....	36
Figure 3.8. PHY module .....	36
Figure 3.9. Radio medium module .....	37
Figure 3.10. Host structure .....	37
Figure 3.11. Transmitter PU model at MAC layer .....	41
Figure 3.12. Receiver PU model at MAC layer.....	41

Figure 3.13. PHY layer coordination with MAC layer.....	46
Figure 4.1. PU host structure in crSimulator .....	47
Figure 4.2. Modified model for PU host.....	48
Figure 4.3. Modified MAC layer for transmitter PU .....	50
Figure 4.4. SU host structure .....	51
Figure 4.5. RTS /CTS mechanism at MAC layer of transmitter SU .....	52
Figure 4.6. RTS /CTS mechanism at MAC layer of receiver SU.....	53
Figure 4.7. Cooperation of MAC layer and spectrum sensor .....	56
Figure 4.8. CRN model in crSimulator.....	57
Figure 4.9. CRN model with two channel .....	58
Figure 5.1. CRN model under single channel.....	59
Figure 5.2. Number of transmitted and received packets with SU average send interval 6ms, PU average send interval 12ms.....	63
Figure 5.3. Number of transmitted and received packets with SU average send interval 12 ms, PU average send interval 6 ms .....	64
Figure 5.4. One channel CRN model with SU relay node.....	66
Figure 5.5. Number of transmitted and received packets SU average send interval 6 ms, PU average send interval 12 ms, with relay.....	67
Figure 5.6. Number of transmitted and received packets SU average send interval 12 ms, PU average send interval 6 ms, with relay.....	68
Figure 5.7. Number of PU and SU successful transmission with constant idle duration ....	73
Figure 5.8. Number of PU and SU successful transmission with constant exponential idle duration .....	74

Figure 5.9. Number of PU and SU successful transmission with varying idle duration .....77

Figure 5.10. Number of PU and SU successful transmission with varying exponential idle duration .....78

Figure 5.11. Two channel CRN model with SU relay node .....79

Figure 5.12. Number of successful transmission for PU and SU with constant idle duration under relay condition ..... 81

Figure 5.13. Number of successful transmission for PU and SU with varying idle duration under relay condition ..... 82



## LIST OF TABLES

Table 2.1. MAC Parameters for 802.11b.....	27
Table 5.1. Parameters in the single channel simulation.....	60
Table 5.2. Parameters in the two channel simulation .....	69



## LIST OF SYMBOLS/ABBREVIATIONS

ACK	Acknowledgment
APP	Application
CSMA/CA	Carrier sense multiple access / collision avoidance
CRN	Cognitive radio network
CSFD	Cyclo-stationary feature detection
CTS	Clear to send
CW	Contention window
CW <sub>max</sub>	Contention window maximum value
CW <sub>min</sub>	Contention window minimum value
DCF	Distributed coordination function
DIFS	Distributed inter frame space
ED	Energy detection
EOT	End of the transmission
FCC	Federal communications commission
GUI	Graphical user interface
IP	Internet protocol
MAC	Medium access control
MF	Matched filter
Mixim	Mixed simulator
MRF	Markov random field
NACK	Negative acknowledgment
NAV	Network allocation vector
NED	Network description file
NS	Network simulator
OMNET++	Objective modular network test-bed in C++
OPNET	Optimized network engineering tools
PHY	Physical
PU	Primary user
RF	Radio frequency
RTS	Request to send
SDR	Software defined radio

SIFS	Short inter frame space
SNR	Signal to noise
SU	Secondary user
TCP	Transmission control protocol
UDP	User datagram protocol
WRAN	Wireless regional area network



## 1. INTRODUCTION

Wireless network users communicate using the radio spectrum. Spectrum is divided into bands and assigned to specific users which are called licensed users to prevent radio interferences. Due to having a license, licensed users have the right to communicate in the frequency band reserved for them. However, when licensed users do not use these bands assigned to them, the spectrum may become idle. Thus, temporary holes called spectrum holes or white spaces may occur in the spectrum.

The use of wireless devices has been increasing in recent years. In parallel with this, new technologies cause the spectrum to become increasingly crowded and spectrum scarcity. The available spectrum is limited so it is necessary to develop new technologies for efficient use of spectrum. In fact, problem of spectrum scarcity arises not because the frequency spectrum is insufficient, but because the spectrum is not fully utilized. Telecommunications authority Federal Communications Commission (FCC) in US has shown that up to 90 percent of licensed bands are not utilized [1].

The measurement of spectrum usage taken from a particular area is shown in Figure 1.1 [2]. It has been seen that the density of spectrum usage is high at some points, while there are also some intervals where spectrum is not used. It seems that the spectrum is not used efficiently and evenly.

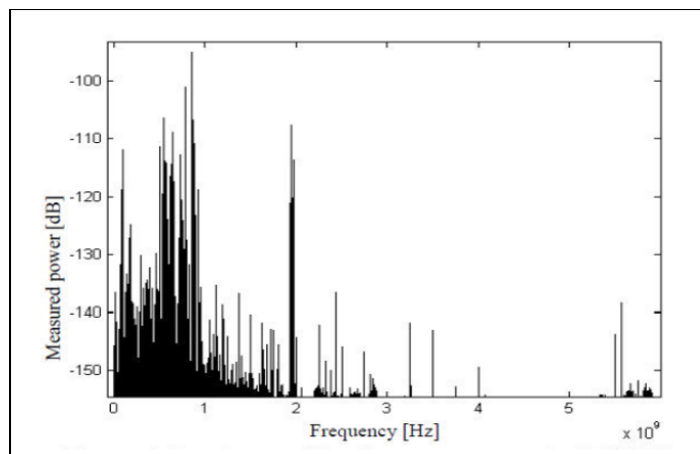


Figure 1.1. Spectrum utilization

The concept of cognitive radio has been developed to enable more efficient use of the spectrum by dynamically detecting the spectrum. CRN aims to increase the spectrum efficiency [3]. To enable this, unlicensed users are allowed to use holes in the spectrum. Based on these, the basic requirements of cognitive radio are to detect the presence of licensed users and to detect the unused spectrum.

Figure 1.2 [4] shows the spectrum where the licensed users transmit and the unlicensed users that move in the spectrum holes where there is not transmission. While the unlicensed users transmit on the spectrum that they find unused spectrum, they switch to the other spectrum hole in case of the arrival of the licensed users.

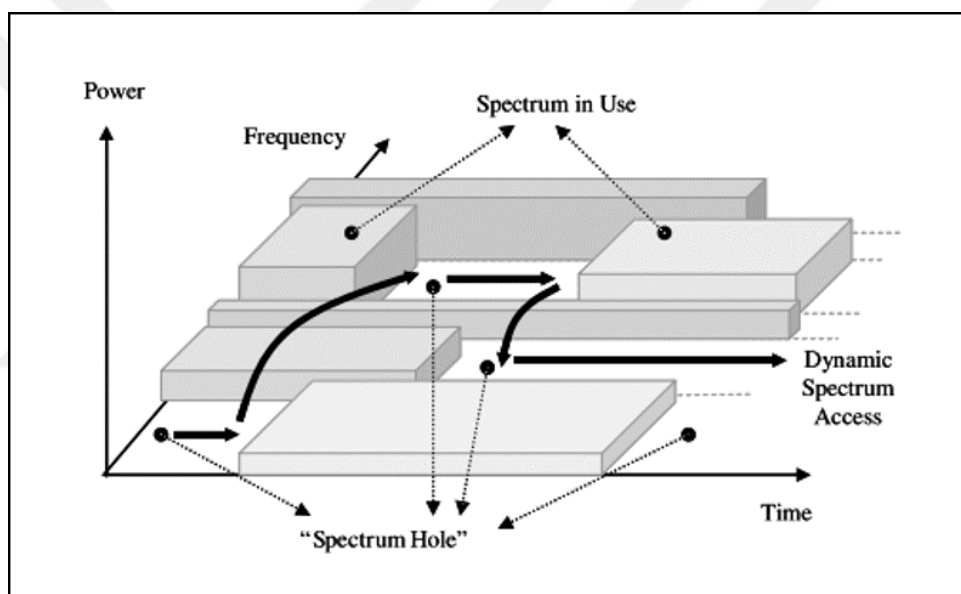


Figure 1.2. Spectrum hole (White space)

In CRNs, two types of users are defined as PU and SU. The priority of using the licensed bands is given to the licensed PUs. On the other hand, unlicensed SUs are able to access the spectrum regions of licensed PUs and can communicate through licensed bands if they do not prevent communication of PUs. SUs must leave the licensed band when PUs start using the spectrum band and must not disrupt communication of the licensed PU.

Since cognitive radios utilize the unused licensed spectrum, spectrum scarcity and the underutilization of the licensed spectrum bands are reduced.

Overlay, underlay and interweave networks define the types of spectrum sharing in CRN [5]. These three approaches allow SUs to access the spectrum in CRNs.

In underlay networks [6], SUs can share the resources of the PUs at the same time as long as interference caused by SU is below some threshold limit value. SUs are allowed to transmit their packets in the licensed spectrum band when PUs are also transmitting.

The overlay system allows SUs to overhear the transmissions of the PUs [7]. SU has information about PU transmission and helps PU transmission. Thus, service quality of PU transmission is increased. The SU prevents interference caused by the PU on the receiver SU. SU also uses some power for its own transmission.

The interweave type of cognitive networks, which is also the scope of this thesis, is the main idea that emerges due to the inefficient use of the spectrum. The spectrum hole in both time and frequency space that PUs do not use can occur. SUs can detect this unused spectrum hole in a certain time period and can communicate there. When the licensed owner of this spectrum comes, the SU leaves from the band and passes to another unused band. In this way, the main aim, which is the efficient use of the spectrum, is ensured [8].

Spectrum sensing, spectrum decision, spectrum sharing and spectrum mobility functions of cognitive radio allow cognitive users to use the spectrum in the cognitive cycle as seen in Figure 1.3 [9]. Unlicensed users must use these functions to access spectrum bands. The cognitive radio user must first perceive the spectrum holes called spectrum sensing process to begin the cognitive cycle. SU periodically checks the PU activity by sensing the spectrum bands. Transmitter SU detects the presence of PU transmission. When there is more than one spectrum hole, SU selects the best spectrum for itself with the spectrum decision function according to the service quality it wants. If the licensed owner of the spectrum comes after the SU starts using the spectrum, the SU must leave the spectrum without interrupting its transmission. To do this, it senses another spectrum hole and continues its transmission by migrating to there. This is spectrum mobility function of cognitive radio. Effective use of spectrum holes between SUs is possible with spectrum sharing function. This function provides coordination among different SUs to access available spectrum holes.

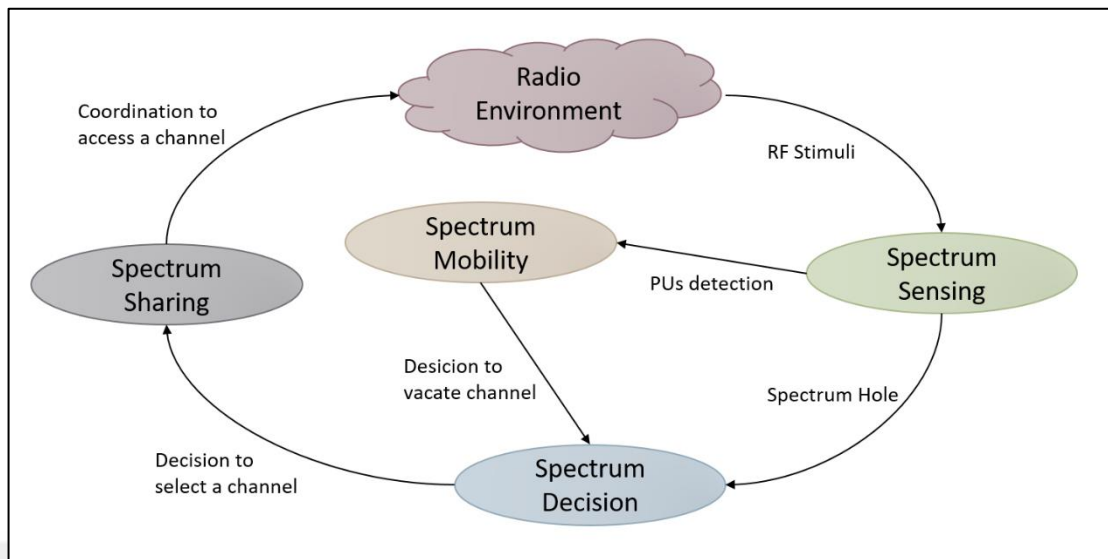


Figure 1.3. Cognitive cycle

## 1.1. LITERATURE SURVEY

There is a great body of research in spectrum sensing and resource allocation in CRNs. In [10], the main idea and functions of cognitive radio with network architecture and application areas were examined.

For the SU to transmit, it must be aware of the PU presence by performing spectrum sensing. Various spectrum sensing techniques have been developed for CRN which are matched filter (MF), energy detection (ED) and cyclo-stationary feature detection (CSFD) [11]. The MF technique senses the spectrum by looking at the signal to noise ratio (SNR). At low SNRs, the MF performs well for spectrum detection by increasing the SNR of the signal. Based on the lack of prior knowledge about the PU in the MF, blind estimation was done using a detector for each PU [12]. It was demonstrated that this study showed better results than ED. ED is the most preferred method in cases of lack of information for PU. ED is a simple method, it detects the presence of PU when it receives the PU signal but the performance of ED is poor at low SNRs [13]. While CSFD can detect the presence of PU at low SNR, it requires prior signal information of the PU. If the periodic amplitude of the PU is known, it makes the most accurate detection. However, it is a difficult method to calculate and to perform in practice [14]. In [15], a new technique was presented to achieve better spectrum sensing performance by combining ED and CSFD methods. Spectrum sensing, one of the

main function of cognitive radio, was developed also with machine learning algorithms to increase accuracy of PU activity [16].

There are researches on different spectrum sharing techniques in order to use the spectrum in the most efficient way. Different techniques have been developed for different purposes. The classification is gathered under four main titles: utilization of spectrum, network architecture, spectrum allocation behavior and spectrum access technique [17].

Spectrum utilization is based on spectrum sharing of licensed and unlicensed users.

Distributed spectrum sharing and centralized spectrum sharing are types of network architecture [18]. If all SUs are managed from a single control mechanism, it is the centralized spectrum sharing. However, if each SU has a separate spectrum sharing mechanism within itself, distributed spectrum sharing takes place.

Cooperative and non-cooperative spectrum sharing take place as spectrum allocation behaviors. In cooperative spectrum sharing, SUs can access the spectrum of PUs. SUs make spectrum allocation by exchanging information among themselves. This technique creates security issues. In [19], SUs acting as relays were used as cooperative sharing to serve the PU in transmission. The PU allowed SUs to use its channel, which would benefit its own transmission at the highest efficiency. SUs did not share information among themselves, they acted individually in non-cooperative spectrum sharing [20]. [21] focused on resource allocation in detail for CRNs and examined allocation types with respect to energy consumption, throughput efficiency, quality of service requirements and interference.

SUs utilize the spectrum according to overlay, underlay and interweave spectrum access techniques. To increase efficiency, underlay and interweave spectrum sharing were combined together [22,23]. In [24], underlay and interweave techniques were combined. The proposed method was compared with interweave and underlay methods separately. It was shown with the results that the presented method increases the throughput of network.

In addition, several approaches for multiband spectrum access have been investigated. In multiband CRNs referring to the future of cognitive radio [25], research was carried out to increase the network efficiency by allowing SUs to collect information about channels and reach to available multiband. [26] presented overlay access model with multiple channel CRN implementation based on Software Defined Radio (SDR). The channel allocation

among SUs was realized with Markov Random Field (MRF). It provided distributed system for SUs with three lower protocol layers. They evaluated the presented model by comparing it with two different resource allocation algorithms in terms of percentage of transmission and collision.

Different types of MAC protocol have been studied to provide medium access control between SUs in the CRN. Wireless standardization studies for spectrum access have been carried out by IEEE. IEEE 802.11 has been provided coordination and control the access to the medium with CSMA/CA [27].

In [28], SU node was developed in OMNET++ that could switch between channels based on the IEEE 802.11 protocol. The authors computed the size of the overlapping area between the PU and the SU coverage and determined the ratio of primary receivers that potentially experienced interference.

The most important commercial standard for medium access in CRN is IEEE 802.22 [29]. MAC protocol in IEEE 802.22 provides spectrum sharing in Wireless Regional Area Network (WRAN) and controls to initialize transmission so IEEE 802.22 has been suitable for CRN.

In paper [30], CSMA/CA in IEEE 802.11 was modified to make it suitable for IEEE 802.22. Access control, the Request to Send / Clear To Send (RTS / CTS) frame structure and the backoff mechanism in CSMA/CA module were modified to use spectrum more effectively and throughput results were examined.

There have also been various studies for design and implementation of CRNs, which is the scope of this thesis. Various CR network simulations such as the Network Simulator (NS) - 2, NS-3, Optimized Network Engineering Tools (OPNET), NETSIM and OMNET++ were examined and these simulation tools were compared to provide researchers decide proper simulation tool for their studies [31].

In [32], CRN model was developed according to the IEEE 802.22 on NETSIM. The CRN included base stations connected to cognitive radio users. Simulations were taken to analyze number of transmitted and collided packet and throughput with changing simulation time.

The Mixed Simulator (Mixim) model was developed to simulate wireless networks in OMNET++[33]. Mixim provided two modules which were physical (PHY) and MAC modules for the SUs.

In the OMNeT++ based cognitive radio models inspired by Mixim [34], the authors modelled a cooperative spectrum sensing mechanism for SUs by adding sensing and clustering modules. In the study presented in [35], the authors developed Mixim node by adding MAC module and introduced the multi-radio model for SUs. They kept PU structure simple. They simulated their model with two different MAC protocols to verify the proposed CR model.

The work presented in [36] introduced a discrete cognitive radio simulation model called crSimulator. It was implemented inspired from Mixim with OMNET++. SU nodes included three layers which were application (APP), MAC and PHY layer with spectrum sensor module and radio frequency (RF) spectrum module. It contributed to SU node with fully protocol stack. It assumed that there were two PU channels and the SU switched to a new channel if it found the current PU channel was busy. SU first scanned the PU channels and transmitted in the absence of any PU activity.

In paper [37], CRN model built under OMNET++4.6 and INETMANET 2.0 environment. PU, SU nodes and modules were based on the existing modules from INETMANET framework in OMNET++ platform. Aim was to validate that PU had more priority right on a channel than SU and SU only accessed a channel when PU did not exist. PU had independent channels with two state idle and busy and it was modelled by Markov chain model. It was proved with simulation that SU utilized the channel without interfering in the absence of PU.

Furthermore, in [36] and [37] PHY and MAC layers were implemented for CRN. Also in [37], SUs transmitted when the channels were not used by PUs. On the other hand, both of the previous works in [36] and [37] did not model the protection of PU transmission under SU activity, since a PU could transmit anytime even if the SU was in transmission. In that case, PU should have the retransmission priority over the SUs.

In [38], two different existing MAC protocols were implemented to crSimulator. CRN was built with one PU and two SUs. Simulations were taken to show how arrival rate of PU

affected throughput. In that study, the receiver PU was not modelled, so end-to-end communication was not performed for PU.

In [34-38], the focus is on the SU transmission. The PU is used as transmitter to keep the environment busy like jammer devices. The receiver PU is not included in CRN, so point-to-point transmission is not provided for PUs.

In the literature, PU protection in CRNs may either refer protection of PU rate constraints under SU activity [39, 40] or design of in advance transmission models so that SU interference to PU transmissions was minimized [41, 42]. In [39-42], the authors aimed to provide a new formulation for protection of PU under SU activity. However, they did not simulate their CRN studies.

In [43], the authors developed a formula to improve the spectrum utilization efficiency of SUs by focusing spectrum sensing, channel selection, energy harvesting and data transmission for the SU. A multi-channel CRN was modelled with ten PU pairs and one SU pair.

In [44], routing protocol with maximum energy efficiency was proposed. The proposed protocol was modelled analytically. Verifying the performance of the proposed protocol with simulation was suggested as future work.

## **1.2. LIST OF CONTRIBUTIONS**

In this thesis, we study to protect PU transmission different from the existing CRN modelling in the related literature. In existing CRN models [34-38], the authors ignore PU transmission. SUs listen to the environment and start transmitting when the channel is not used by PUs. In these studies, the interweave network is not guaranteed as end-to-end transmission for PUs is not provided. For this reason, we add a receiver PU to ensure real interweave CRN. Unlike [34-38], we provide end-to-end transmission for PUs with destination PU node. PUs do not just work like jammers to keep the environment busy, we now provide real communication system with MAC protocol for them as well.

We model interweave-based single-channel and two-channel CRN. We make use of existing medium access control models (CsmacaMac, AckingMac) for our CRN models and modify

these models to make them suitable for CRN. Different from [39-42], we simulate these models, which ensure that PU transmission is protected against SUs.

We design our first CRN model based on INET Framework on OMNeT++. The CSMA/CA based CsmCaMac [45] module offered by INET is suitable for modelling SUs. However, the AckingMac [46] module, which is the other MAC protocol offered by INET, does not provide PU transmission in accordance with the cognitive radio concept and does not work in compatibility with the CsmCaMac module. Thus, it is not suitable to use for PU in CRN. For this reason, new PU module which is compatible with CsmCaMac module and the cognitive radio concept is required. We propose this new MAC module for PU. The PU model has been realized by taking into account the protection of PU transmission, which is our motivation to start this study.

We present a further two channel CRN model based on the crSimulator. In crSimulator, the focus is on SU transmission, point-to-point PU transmission is not provided. Transmitter PU node is used in the CRN with purpose of keeping channels busy. This node keeps the channels busy by sending packets at regular intervals. There is no destination node to which packets are forwarded, and node cannot inform whether the packet has been reached. Due to these missing points, we develop a new PU model for our two channel CRN.

In both PU models developed, we provide end-to-end communication for PUs with transmitter and receiver nodes in accordance with cognitive radio logic. PU nodes have an acknowledge and retransmission mechanism with fully developed protocol stack to define a real network. We ensure that the presence of SU has no effect on PU transmission.

Along with these improvements, we have also improved INET and crSimulator to conform to CRN simulation with PU communication. Thus, we offer simulation models that support PU transmission, which enable simulating various cognitive radio studies.

### **1.3. THESIS ORGANISATION**

We organize the rest of the thesis as follows.

In Chapter 2, we provide fundamental information about OMNET++ and MAC protocols we use.

In Chapter 3, we describe the features and implementation of our single-channel CRN model.

In Chapter 4, we describe the features and implementation of our two-channel CRN model.

In Chapter 5, we present simulations with numerical results and discussions for both CRN models.

In Chapter 6, we summarize our developed CRN models with simulations and their results. In addition, we address future research directions.



## **2. PRELIMINARIES**

In this section, we give information about the OMNET++ simulator program that we used while creating the CRNs. Then, we describe the IEEE 802.11 MAC protocol we use in CRNs, together with its techniques.

### **2.1. OMNET++**

OMNET++ [47], an object-oriented and modular discrete event-based network simulator, is used in this thesis. OMNET++, an open source simulator, allows modeling of new communication networks by using C++ programming language. This simulator tool provides researchers to design frameworks such as protocols, mobility, wireless sensor networks.

Network topology definitions are performed by creating .ned extension files using Network Description File (NED) language. It allows network topologies to be designed visually. Simulated objects are represented by modules. Modules can be simple or combined. Modules communicate through messages sent directly or through gates. Simulations can be easily configured using an .ini file.

Modules in simulation exchange their messages to communicate with each other at discrete time. OMNET++ library presents modules to researchers like Mixim mobile and wireless simulation, INET Framework, Transmission Control Protocol / Internet Protocol (TCP/IP) based simulation. By inspired from this, new modules can be created. It enables to debug, trace animation with Graphical User Interface (GUI) and take graphical results of simulation. INET Framework is most developed framework until now which includes all protocols in it.

### **2.2. MAC PROTOCOL IN IEEE 802.11**

The MAC protocol is required to perform the CRN functions. The MAC layer defined in the IEEE 802.11 protocol is used in the CRN design for channel detection and channel access management [48]. Distributed coordination function (DCF) is a protocol used in wireless networks where any user can access and transmit randomly.

In IEEE 802.11, two access methods are defined in the DCF protocol. These methods are CSMA/CA and RTS/CTS.

### **2.2.1. CSMA/CA Technique**

The MAC protocol defined in the IEEE 802.11 standards provides the coordination of the nodes communicating on the common channel [48]. The CSMA/CA technique, which is the main access method of the MAC layer, provides collision avoidance in wireless networks. The CSMA/CA has been created to detect a collision in wireless local area networks. "Listen before talk" principle in the CSMA/CA is used to understand if a node transmits its packet or not. Thus, nodes can sense a channel before it starts to send its packet. The acknowledgment mechanism in the CSMA/CA is used to determine whether the packet sent from the transmitter node has successfully reached the receiver node. If the packet reaches the receiver node successfully, the node sends an acknowledgment to the transmitter node to report it after a short time interval. When transmitter node does not receive acknowledgment, the packet is considered as lost and same packet transmission occurs again called retransmission.

Distributed Inter Frame Space (DIFS) is the time the node waits to prevent a collision in the channel. After transferring the packet, the transmitter node waits for the ACK packet from the receiver node. If the ACK packet arrives at the transmitter node, the transmission has taken place without collision at the receiver node. If the ACK packet is not received, a collision has occurred, and the transmitter node will send the same packet again.

The backoff algorithm avoids contention between nodes at the same medium. To access the medium, a random time frame is given to each node. The random backoff method reduces the probability of the transmission start time being the same between the nodes and prevents them from transmitting at the same time.

The backoff time is calculated as a multiple of a random value between 0-CW called the Contention Window. The minimum and maximum values of CW are assigned as 31 and 1023, respectively. The CW value doubles in every retransmission state. If the transmission is successful after retransmissions, it takes the minimum value of CW (CW<sub>min</sub>). MAC parameters in 802.11b are shown in the Table 2.1 [48].

Table 2.1. MAC Parameters for 802.11b

Parameters	SIFS( $\mu$ s)	DIFS( $\mu$ s)	Slot( $\mu$ s)	CWmin	CWmax
802.11b PHY	10	50	20	31	1023

Figure 2.1 shows CSMA/CA working mechanism briefly. Transmitter node waits medium to become idle at the beginning of the transmission. After that, the node waits DIFS time. If medium is idle end of the DIFS, backoff algorithm starts. When medium is busy during backoff time, backoff timer stops until medium becomes idle again. If the medium is empty at the end of the DIFS timer, backoff starts with its last value. Packet is transmitted end of the backoff time. Backoff timer starts to decrease until medium becomes busy.

Transmitter node waits ACK from receiver node after it sends packet. If it does not receive ACK, its backoff algorithm starts. The CSMA/CA flow at the transmitter node is shown in Figure 2.2. Receiver node waits during Short Inter Frame Space (SIFS) time to send ACK packet after packet arrives. After the SIFS time, ACK is sent regardless of whether the medium is busy or empty.

DIFS time is longer than SIFS time as seen in the equation (2.1) because it is important to complete transmission by transmitting ACK frame. It prevents the ACK from being lost.

$$DIFS = SIFS + 2 \times Slot\ Time \quad (2.1)$$

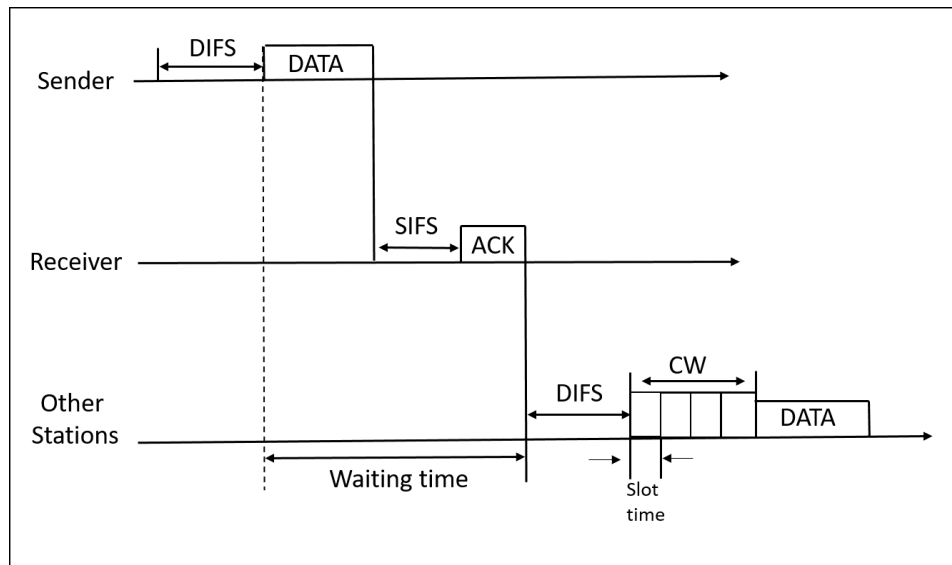


Figure 2.1. CSMA/CA working mechanism

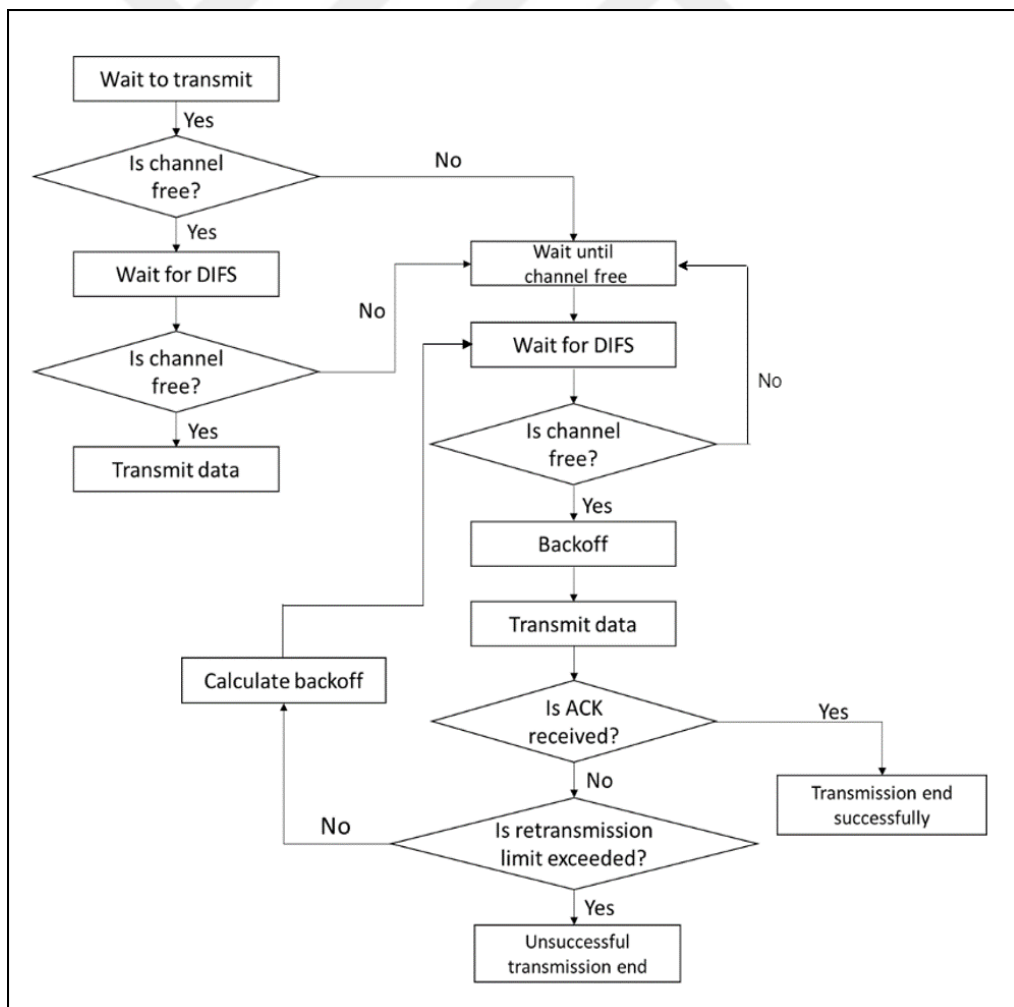


Figure 2.2. Flowchart of CSMA/CA at transmitter node

### 2.2.2. RTS/CTS Technique

The RTS/CTS mechanism is standardized as an access method in DCF. The RTS/CTS mechanism aims to avoid collision and minimize effect of collisions.

In this mechanism, transmitter sends RTS to receiver before it sends its packet. In this way, the receiver is informed that data will be sent soon. After the RTS reaches the receiver, the receiver returns this message with the CTS. The receiver notifies the transmitter that it is ready to receive data with the CTS message.

Figure 2.3 shows RTS/CTS mechanism at transmitter node. At the beginning of the transmission, the transmitter first listens to the medium. If the channel is free, it starts the packet forwarding at the end of the DIFS time. On the contrary, if the medium is not empty, it waits until it is empty. If the medium continues to be empty for the backoff period after the DIFS period, it transmits RTS. When the RTS is received by the receiver, the receiver waits for the SIFS period and starts the CTS transmission. After receiving the CTS, the transmitter waits until SIFS and transmits its package. After receiving the packet, the receiver waits for the SIFS period and sends an ACK packet.

A collision occurs when a CTS is not received in response to the RTS in the transmission. In a transmission, when RTS messages are sent by more than one node at the same time, these RTS messages might collide. This causes the receiver not to receive RTS and no CTS message to be sent to transmitter. When the transmitter does not receive CTS, it perceives this as collision. It attempts to initiate the backoff mechanism and tries to start the transmission again.

Network Allocation Vector (NAV) is defined for users to avoid accessing the medium. When RTS and CTS control frames are sent, other users in transmission range also receive these control frames. Thus, users assume transmission and enter waiting time which is called NAV as seen in Figure 2.4 [49]. NAV gives information about how long the medium will not be used by other users. The transmissions of other users stop until ACK is sent.

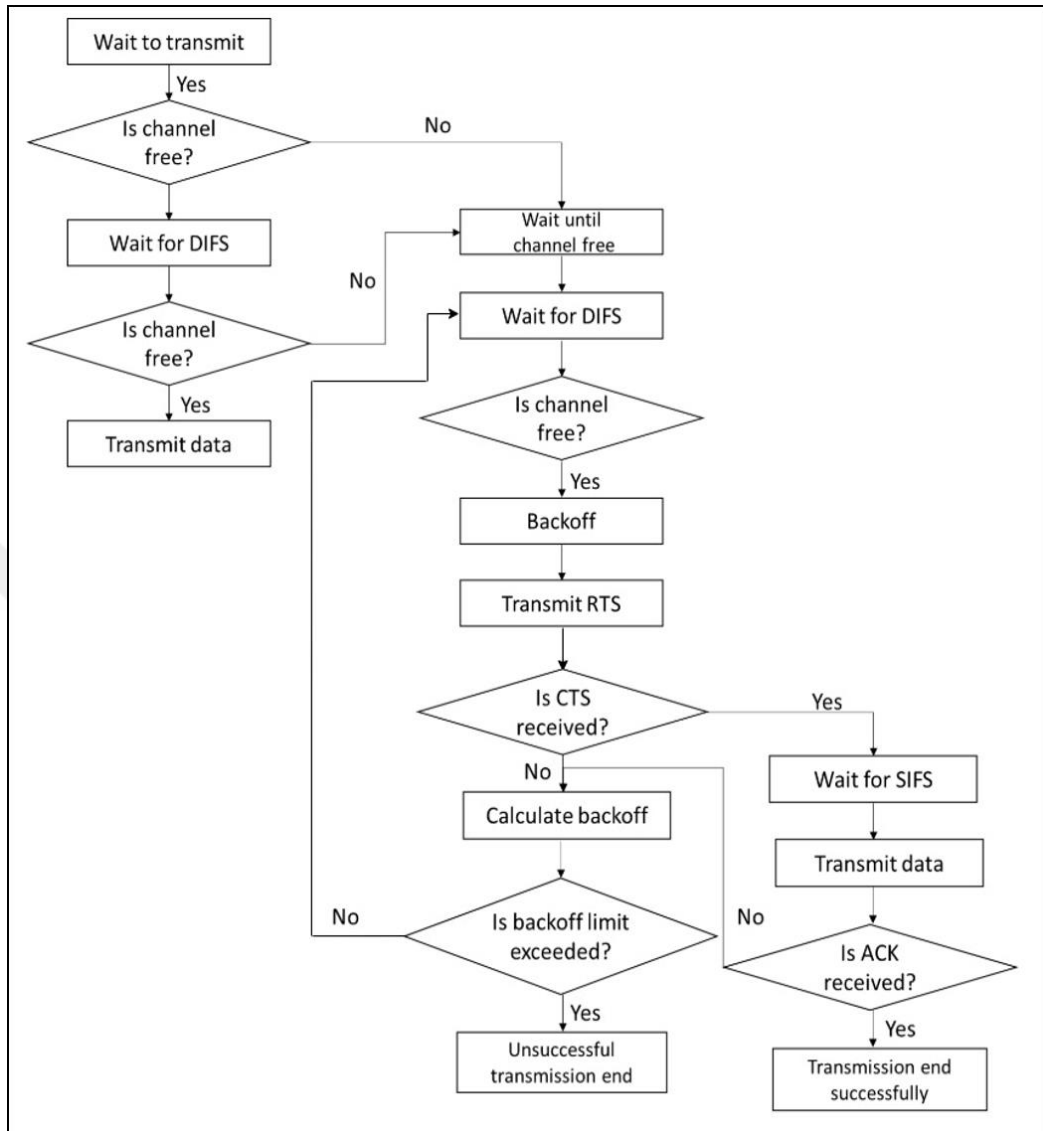


Figure 2.3. Flowchart of RTS/CTS at transmitter node

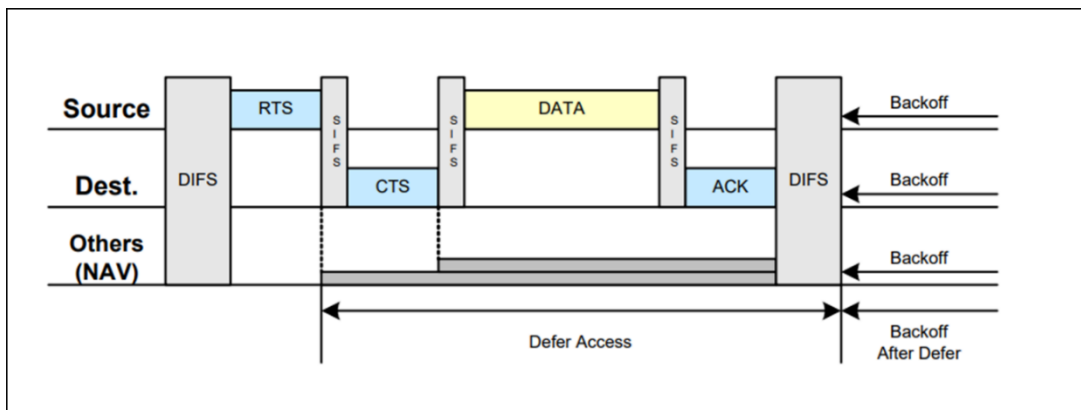


Figure 2.4. RTS/CTS and backoff mechanism of DCF

### 2.2.3. Backoff Algorithm

Backoff is an algorithm used to bring solution to the contention between the nodes defined in the DCF. It can also be expressed as an additional time used after DIFS. During transmission, the algorithm reduces collisions of packets sent from different nodes.

Every node has a randomly selected backoff time in the range  $(0, CW)$  after duration of the DIFS period. Backoff algorithm in IEEE 802.11 has been shown in Figure 2.5 [50]. At the beginning of the first time, all nodes set the CW value as  $CW_{min}$ . At the end of the backoff period, the node that performs transmission, sets the CW value to  $CW_{min}$ . However, if the medium is still busy at the end of the backoff time, the node does not transmit freezes the backoff time until the medium is empty. If the medium is empty and there is frozen backoff time, it is waited backoff time. Otherwise, if there is no frozen backoff time, a new backoff time is set in the range  $(0, CW)$ . If the transmission is terminated unsuccessfully and retransmission is necessary, CW is updated with  $2CW$  until CW is equal to maximum value of CW ( $CW_{max}$ ).

If the backoff times of the nodes end at the same time, they can start transmitting at the same time when the medium is empty. In this case, a collision occurs. After collision, it requires longer time for colliding nodes to start transmitting due to the doubled CW.

Freezing the backoff time in the backoff algorithm allows the node that has tried to transmit before and cannot transmit due to busy medium to enter the medium earlier than the node that wants to transmit after this node.

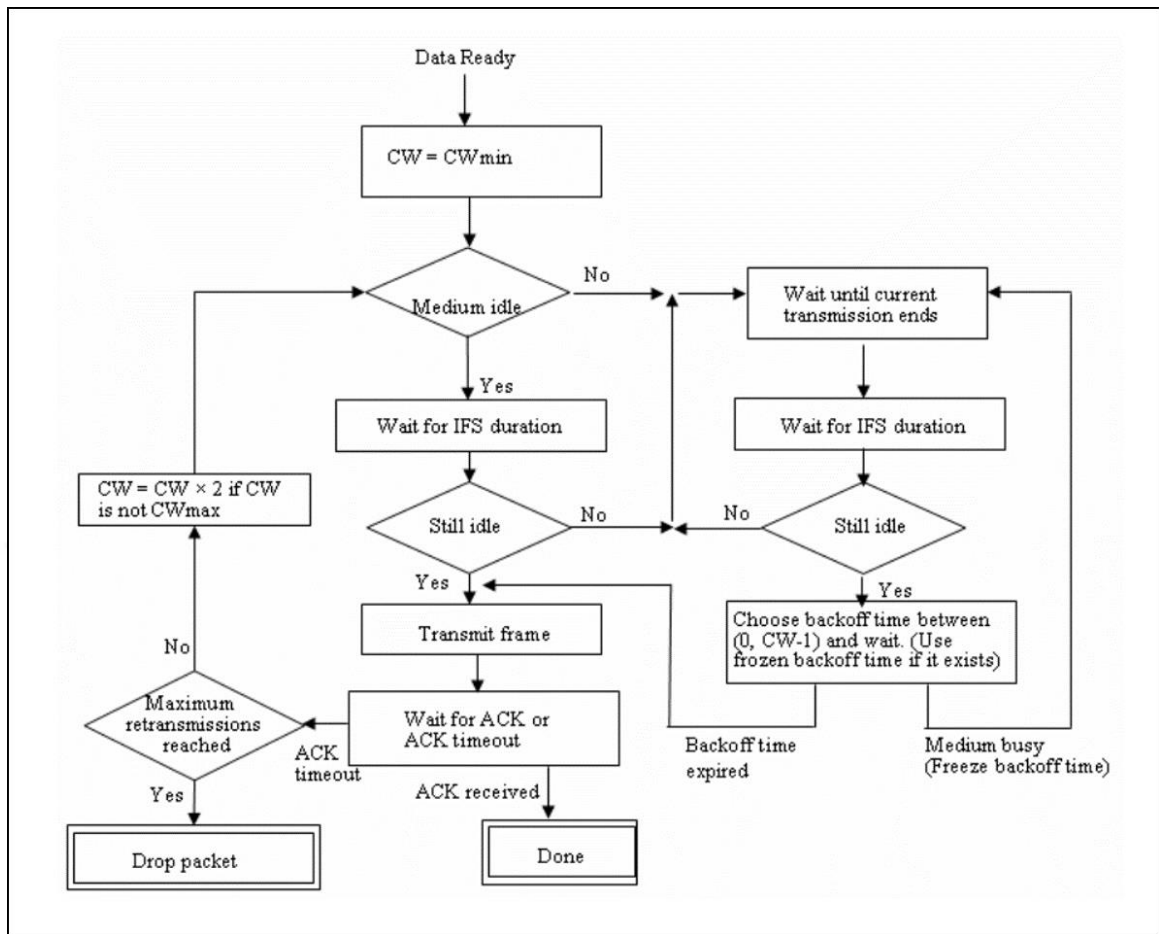


Figure 2.5. Backoff algorithm in IEEE 802.11

### 3. CRN MODEL UNDER SINGLE CHANNEL

In this section, we explain our CRN model with one channel that we have implemented. We develop a single-channel simulation by taking into account the rights of the PUs.

First CRN model is designed by modifying INET 4.2.1 using Omnet++.

#### 3.1. INET FRAMEWORK

INET Framework is a network simulator with an open source library built on OMNET++. INET includes various protocols and interfaces in it. Researchers can use existing components to create new models.

##### 3.1.1. Architecture of Host

The implementation of the nodes is carried out using the compound modules available in INET.

**Host module:** WirelessHost module in Figure 3.1, which is 802.11 card, is used for all CRN nodes.

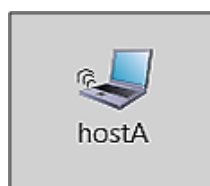


Figure 3.1. Host module in CRN

Protocol layers of CRN host consist of PHY layer, Link layer, Network layer, Transport layer and APP layer.

**Application Layer:** User Datagram Protocol (UDP) packets are sent to the given IP address at the given interval from APP layer and hosts in receiver side discard packets at APP layer. UdpBasicApp simple module in Figure 3.2 is used to generate UDP packets in transmitter node. This module also allows us to set packet length in bytes, source and destination

addresses and port numbers of hosts and packet send interval. The time interval of sending packets is adjusted using the random exponential algorithm.

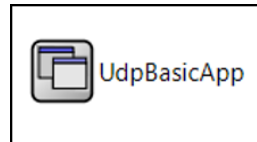


Figure 3.2. APP module in CRN

On the contrary, UdpSink simple module in Figure 3.3 allows receiver host to discard UDP packets at APP level. Also, it provides to count received packets.



Figure 3.3. UDP Sink module in CRN

All configurations about APP layer are set in the .ini file.

**Transport layer:** Figure 3.4 shows the Transport layer module which supports TCP, UDP protocols to transfer packet between Network and APP layer.



Figure 3.4. Transport layer module in CRN

**Network layer:** IP address is automatically assigned to all interfaces in the network and static routing is set up for each host in Network layer. Figure 3.5 represents the Network layer module.

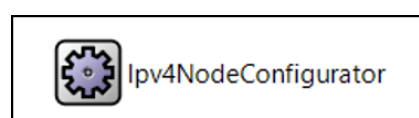


Figure 3.5. Network layer module in CRN

Each node has information of the interface table. Thus, static routing allows packets to be routed according to the routing table when a host is used as a relay. Also, the interface table is recorded in the Link layer. In this way, MAC addresses of the hosts are known at each node. If the packet arrives the host that is not addressed to it, Link layer drops the packet.

**Link layer:** The MAC interface module is used to implement the MAC protocol. The module together with the radio interface module as seen in Figure 3.6 is implemented in the wireless interface compound module.

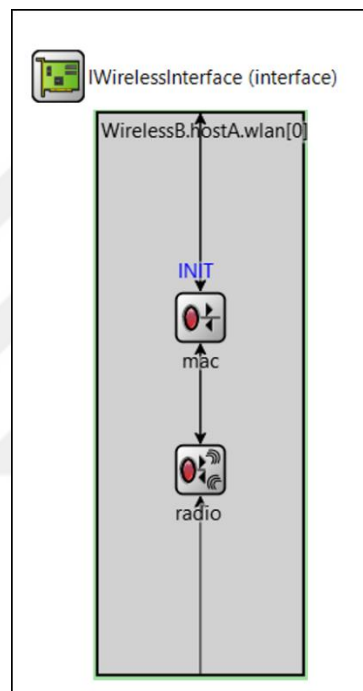


Figure 3.6. Radio / MAC module

CSMA MAC protocol is used at Link layer. CSMA module allows host to sense channel availability. Packet retransmission and acknowledgment functions are also performed as MAC protocol. Our CRN model is developed by modifying CSMA module of the existing INET Framework. This is explained in the later part.

The MAC module is supported by the queue module interface. This module interface ensures that the packet comes from the upper layer is kept in the MAC queue in Figure 3.7 before it is sent to the lower layer.

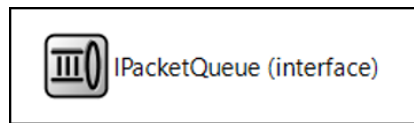


Figure 3.7. MAC queue module

**Physical layer:** PHY layer is responsible for setting parameters of communication range and interference range. Successful transmission between transmitter and receiver hosts depends on communication range and interference range.

The radio interface module in Figure 3.8 receives the frames from the upper layer and sends these frames as signal messages to other nodes in the communication range. It transmits the signal from other nodes to the upper layer as a frame. The radio interface module is used in the wireless interface compound module.

Configuration of communication range and interference range are done in .ini file.

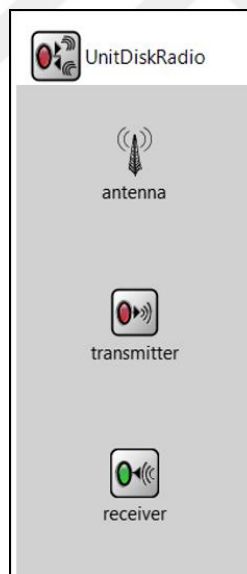


Figure 3.8. PHY module

**Radio Medium:** The medium in which the transmission takes place is defined as the Radio Medium module in Figure 3.9. It makes transmission interfering in the medium.

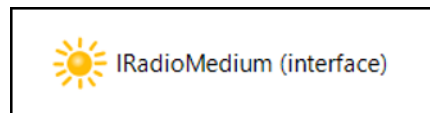


Figure 3.9. Radio medium module

Figure 3.10 shows the host structure used in single channel CRN with layers.

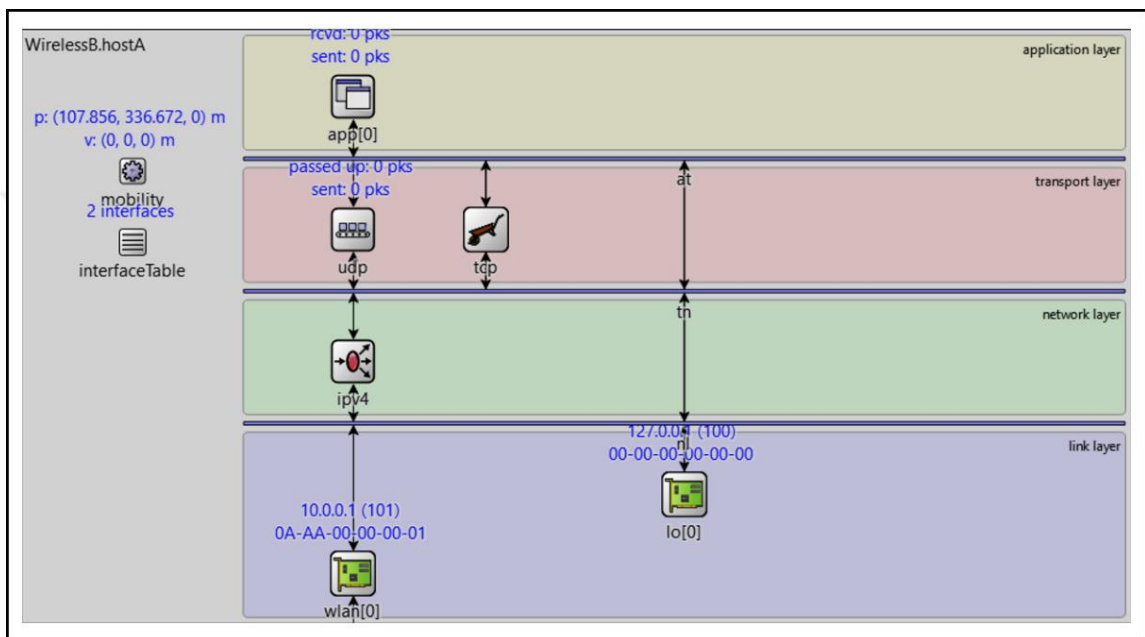


Figure 3.10. Host structure

### 3.1.2. MAC Modules for Wireless Communication in INET

While creating the first CRN, our goal is to create an interweave network that allows users to communicate through a single channel. While PUs with the license are not transmitting, SUs must detect that the medium is empty and begin transmitting. Then, when the PU starts transmitting again, the SUs are expected to leave the medium. An acknowledge system should be created so that users can understand that there is a packet loss when there is a collision and a retransmission system should be created so that the packet can be retransmitted in the case of possible packet loss. However, since the priority of the transmission right is intended to be for the licensed users, it is aimed that the PUs have the right to use the medium first with retransmission, in case of a possible collision. While PU

is able to initialize its transmission for retransmission after collision, SU can retransmit its colliding packet, only if it senses the channel as empty.

INET Framework provides two modules as MAC protocol which are AckingMac and CsmacaMac for wireless communication. In AckingMac module, users transmit to channel immediately when they have packet to send without first listening the medium. Thus, AckingMac may seem appropriate to model the PUs, since they have right to transmit anytime. However, AckingMac does not support retransmission for colliding packets, which does not ensure PU completes the transmission successfully. Moreover, the module cannot be used for MAC protocol of SUs. Because in cognitive radio, unlicensed users cannot transmit whenever they want.

The CsmacaMac module functions as a protocol based on IEEE 802.11b. Since it is a CSMA/CA based MAC protocol, it has acknowledgment algorithm, backoff algorithm, DIFS and SIFS timing and retransmission. Thus, CsmacaMac module provides necessary features to sense channel availability for unlicensed users in the cognitive radio and it is convenient to use the existing module for SUs. However, it is not suitable for PU. Because the license owner user does not need to wait for the availability of the channel to start the transmission. Therefore, the backoff algorithm, DIFS and SIFS timing are not required. In addition, AckingMac for MAC protocol of PU host is wanted to use together with CsmacaMac for SU host, when a packet is forwarded from one module to another (PU and SU can receive each other's packet up to the MAC layer), a mismatch occurs due to the different header types of the modules. Thus, AckingMac and CsmacaMac cannot be used together to create CRN in INET 4.2.1.

Thus, it has been required to develop a new module for PU which is compatible with the header type of CsmacaMac and supports retransmission and acknowledgment without waiting time like DIFS and backoff.

### **3.2. PU MODIFIED MAC LAYER**

For the reasons explained in the previous section, new MAC model has to be designed for PUs. In the new model, the PU must transmit the packet without listening to the environment, and if the packet cannot be successfully transmitted to the receiver, same packet must be

sent again, so PU must have acknowledgment and retransmission mechanism. Moreover, PU must have the first retransmission right when the collision occurs.

In transmission, packets are generated from the APP layer according to the send interval range. When packets reach the MAC level, they first wait in the MAC queue. If there is a packet sent from the APP layer in the MAC queue, the PU is allowed to send the packet regardless of the availability of the environment. The only situation that prevents the PU from sending the packet is if the PU is transmitting. The PU can start new transmission only when its last transmission has ended. Thus, the PU has the right to start transmission without sensing the medium as soon as the transmission time comes from APP level or if there is a packet on the MAC queue.

Acknowledgment and retransmission mechanism are also included in the MAC layer of PUs. In this way, it is aimed to prevent packet losses caused by collision.

In the acknowledgment mechanism, ACK timer is set in the transmitter PU. This parameter can be configured in the .ini file. After the packet is transmitted from the MAC layer, the ACK timer is started. Transmitter PU waits for the ACK to come from the receiver PU within the time value entered in the configuration file. If the receiver PU receives the data packet, it sends the ACK packet without waiting. Here, there is no SIFS duration in the PU. Because the receiver PU is not expected to listen to the environment and it has the priority to use the environment. If the transmitter PU receives the ACK packet until the ACK timer period expires, successful transmission is ensured. However, if the ACK is not received within the ACK timer period, the packet transmission is failed due to collision and the packet must be transmitted again. Since the licensed users have the right to use the medium first in cognitive radio, PUs are expected to utilize the environment before unlicensed users by transmitting the collision packet first. After the collision, the PU is allowed to retransmit the collision packet before the SU. For this purpose, waiting time like DIFS and backoff timer are not defined in the retransmission algorithm of PU so PU retransmits its collision packet as soon as it detects a collision. Thus, priority is given to the PU in the retransmission.

The maximum retransmission value is determined for packet retransmission. Retransmission can occur for the same packet up to this value. If there is still no successful transmission at the maximum retransmission value, the packet is dropped and unsuccessful transmission occurs which is resulting in packet loss.

**Transmission Policy:** Figure 3.11 shows the transmitter PU behavior in MAC layer, while Figure 3.12 indicates the receiver PU model. The MAC operation at the PU side is as follows. According to send interval settings of transmitter users, PU or SU starts transmission first. At the beginning, both transmitter and receiver PUs are in IDLE state. If PU starts the transmission first because of its shorter send interval, the packet arrives to Link layer from the APP layer and PU goes into TRANSMIT state. Later, when the packet is sent into the medium, transmitter PU schedules the ACK timer and switches its state as WAITACK. When PU receiver receives packet, it enters RECEIVER state and responds with an ACK. After ACK packet is sent, its state enters to IDLE. If transmitter PU receives ACK successfully, it cancels ACK timer and enters the IDLE state. If transmitter PU does not receive ACK packet in time, ACK timeout takes place. Then, if the retransmission limit is not exceeded transmitter PU enters IDLE state and retries current transmission without waiting. When the retransmission limit for the same packet is exceeded, an unsuccessful transmission occurs for that packet.

When the packet sending interval is longer for the PU than for the SU, the SU transmits its first packet to the medium first. However, as soon as the send time for the PU comes, the PU starts transmitting its packet regardless of the activity of the SU in the medium. Thus, while the SU is sending packet, if time has come for PU transmission, PU also starts transmitting. This causes packets to collide. However, as there is a retransmission mechanism for both type of users, packets can be retransmitted without being lost.

When SU transmits due to empty medium, PU receives the SU packet. When a packet is received from SU, the MAC state of PU switches to RECEIVER. However, packet is dropped since it is not addressed to PU and MAC state of PU goes back to IDLE.

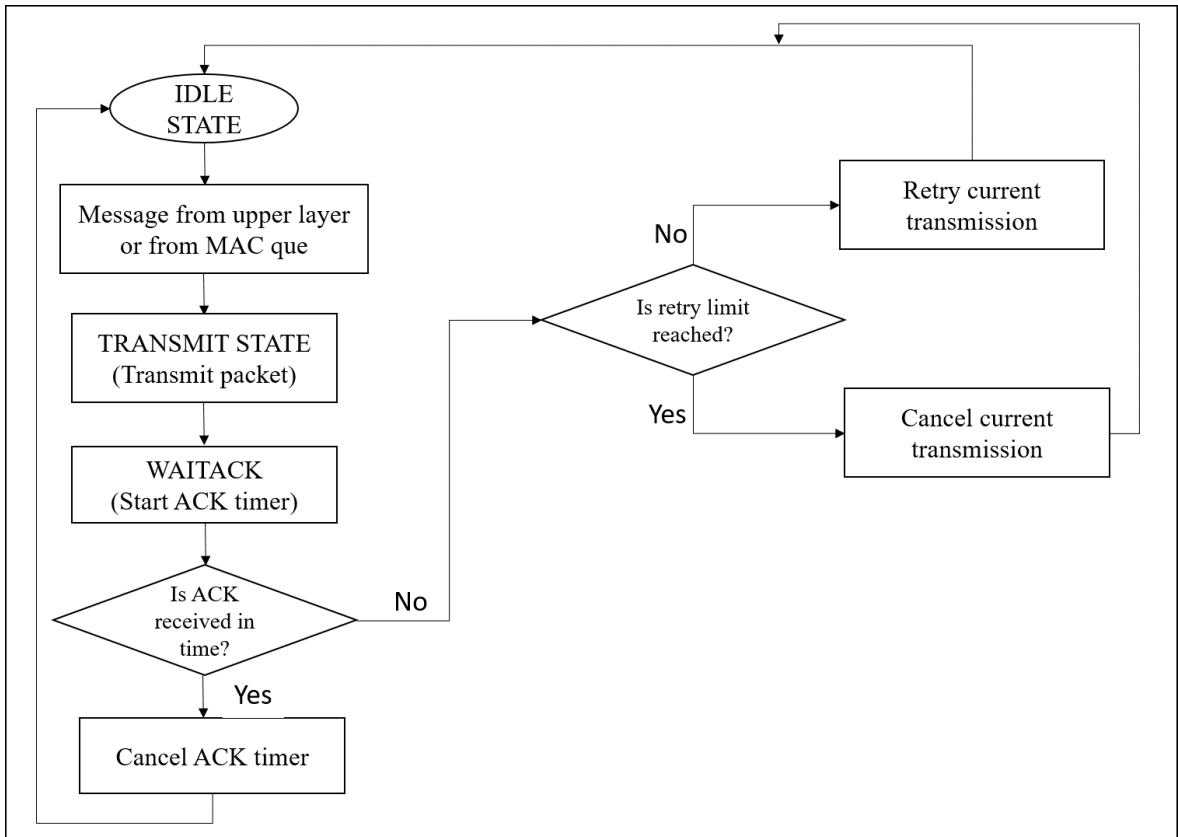


Figure 3.11. Transmitter PU model at MAC layer

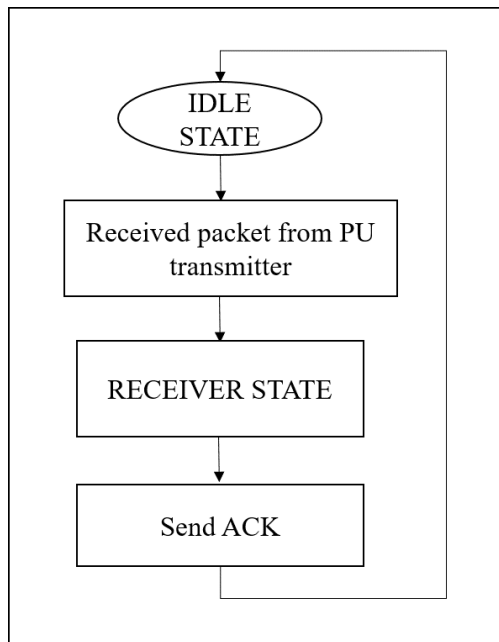


Figure 3.12. Receiver PU model at MAC layer

### 3.3. SU MAC LAYER

When a packet comes to MAC layer of SU from the higher layer, it is first pushed in the MAC queue. Then, the MAC state is checked, if it is in a state other than IDLE, the upper layer transmission is suspended. The packet waits in the MAC queue.

The most important feature for SUs in CRN is that they utilize medium when it is empty. MAC protocol is used to access the medium for SU. It allows SUs to decide when they access to the medium and to use retransmission and ACK mechanism.

The CSMA MAC protocol based on IEEE 802.11b is defined in INET 4.2.1, which works as follows: SU host that wants to transmit senses the medium, if the medium is busy, for example some other host may be transmitting, then the host that wants to transmit postpones this request to a later time. If the medium is perceived as free then the host transmits. Such protocols are very effective when the medium is not over crowded since they allow hosts to transmit with the least delay, but there is always a possibility that hosts may transmit simultaneously which result in collision. In such collision conditions, MAC layer defines packet retransmission. When SU host wants to retransmit its packets to the medium due to a collision, and if the medium is busy, then host delays this request again. If the medium is free for a certain period that is called DIFS, then the host can retransmit packet. On the receiver side of SU, if acknowledgment setting is on and the receiver SU receives a data frame addressed to it, it responds with an ACK frame after waiting SIFS time. If the transmitter SU does not receive ACK packet in time, it retransmits its packet. Backoff is a method to resolve the conflict between different hosts that want to access the medium. The host drops the packet after it tries retransmission a certain number of times.

The described features of this protocol defined in INET make it suitable for SUs. This protocol presents as a CsmCaMac module. In our CRN model, we use the existing CsmCaMac module for SUs. Behaviors of SUs in CRN are created using the CsmCaMac module. The function of SUs in the MAC layer is explained below.

**Transmission Policy:** The SU transmitter senses medium and decides when it can transmit its packet. In receiver side, the SU acknowledges the packets. When transmission starts in the medium, the MAC layer of SU understands that the state of the medium has changed due to the state change in the PHY layer.

Then, it waits for the packet from the medium and switches to the receiver state. When the packet received from the environment reaches the MAC layer from the lower layer, if packet is from PU, the packet is dropped, but if it is from SU, the packet is accepted. After the packet is dropped when it comes from the PU, the MAC state switches back to IDLE and if there is a waiting package in the MAC queue, it is popped from the queue. In this case, after receiving the packet from the upper layer, if there is no transmission in the medium, the DIFS timer starts if the medium is empty. Then, MAC state becomes WAITDIFS. If the medium is reused by PUs before the DIFS expires, the MAC level of SU is notified that the medium is busy so DIFS timer is cancelled. In this case, the SU state is set to the receiver at MAC level. When the medium is empty again, the DIFS timer starts again. After the DIFS time ends, the backoff timer is scheduled and the MAC state switches from WAITDIFS to BACKOFF. When calculating the backoff time, it is set according to the MAC parameters for 802.11b in Table 2.1. A random integer number selected between  $[0, CW+1)$  is set as a slot. Then, backoff is calculated according to formula (3.1).

$$BackoffPeriod = slots \times slotTime \quad (3.1)$$

If the medium is still empty at the end of the backoff period, packet transmission starts and the TRANSMIT state is passed. The MAC level is triggered by PHY layer when last packet transmission is finished. Thus, MAC detects that the transmission is over and enters the WAITACK state and starts the ACK timer to know if the packet reaches the receiver successfully. If ACK is received before the ACK timer ends, the ACK timer is cancelled and if there is a packet in the MAC queue, this packet is popped. If ACK is not received within the ACK timer, it means that there has been a collision in the environment. Retransmission starts for the undelivered packet. The backoff period is set by doubling CW value. After the medium is detected as empty, the DIFS timer starts. If the medium is still empty at the end of the DIFS period, the backoff timer is started. In the meantime, if a transmission is detected in the environment, the backoff is cancelled and the remaining backoff time is preserved. When the medium is empty again, it is waited until this preserved backoff time after the DIFS period. Packet retransmission continues until the configured maximum retry is reached. If there is still no successful transmission at maximum retry, then packet loss occurs.

For the receiver SU, if it receives a packet that is not addressed to it, it drops the packet at the MAC level. When it receives a packet addressed to it, it switches state from RECEIVE

to WAITIFS, and starts the SIFS timer. ACK packet is sent at the end of the timer. The SIFS duration is set as indicated in the Table 2.1.

### **3.4. PHY LAYER**

UnitDiskRadio module in INET is used to model the PHY layer for both PUs and SUs. For CRN model, the PHY layer is kept simple with communication, interference and detection range. Successful transmission depends on communication and interference range. The module includes separate transmitter and receiver model. While packets from upper layer are converted into electrical signals by transmitter, electrical signals from other nodes are converted into packets by receiver.

In the module, radio mode, transmission states and reception states are used for the functionality of the PHY layer. According to these states, it is understood whether there is signal transmission in the medium.

The radio mode is defined to determine how radio should operate. The radio mode is set according to the MAC layer behavior. Radio mode is changed when message (data or ACK) is sent from the MAC layer to the lower layer. Also, when the medium state changes, it is detected at the MAC layer and the radio mode is triggered from the MAC layer according to the changing medium.

Reception state indicates the state of the radio environment. If it is not possible to detect a signal, it is set as undefined. When the medium is not busy, it is set to idle. If a signal can be detected from the environment, it is set as receiver.

For the transmission state, if the radio mode is unknown, it is called undefined. If the last transmission is completed or no signal is being transmitted to the medium, transmission state set to the idle. Transmission state is changed to transmitting while a signal is transmitting to the medium.

For the implementation of cognitive radio, the ability of SUs to perceive the spectrum is the most important issue. SUs must sense the medium to transmit by detecting activity of PUs. Spectrum sensing mechanism is carried out on PHY layer.

The PHY layer in the UnitDiskRadio module works as follows in Figure 3.13. Before the frame is transferred from the MAC layer to the PHY layer, the radio mode is set to TRANSMITTER at the transmitter node. In this way, the radio prepares to transmit frame and reception of frame is not possible at this case. Transmission state of radio is set to IDLE from UNDEFINED so there is no transmitting signal on the radio medium. Reception state of radio is set to UNDEFINED from IDLE because radio mode is transmitter. Then, the signal encapsulated from the frame is sent to other nodes in communication range. The radio transmitted signal part is set from NONE to WHOLE. Then, transmission ends and transmission state of radio is updated from TRANSMITTING to IDLE because the last transmission is over. At this point, the state changing is reported to the MAC layer and the transmission state of radio is checked at the MAC module. If the radio transmission state has changed from TRANSMITTED to IDLE and MAC state is WAITACK, the radio mode is updated as RECEIVER so radio is ready to receive frame. Later, the receiver listens to the environment at radio level. If no transmission is detected, reception state of the radio is updated as IDLE. Radio transmission state and transmitted signal part are changed respectively as UNDEFINED and NONE. If the message is detected at PHY layer, reception starts. If message power is receivable, radio reception state becomes RECEIVING and radio signal part is set to WHOLE. If radio signal part is WHOLE, reception decision is made by looking at interference, SNR. If reception is successful, the signal is decapsulated and transmitted to the MAC frame. Then, it is sent to the upper layer.

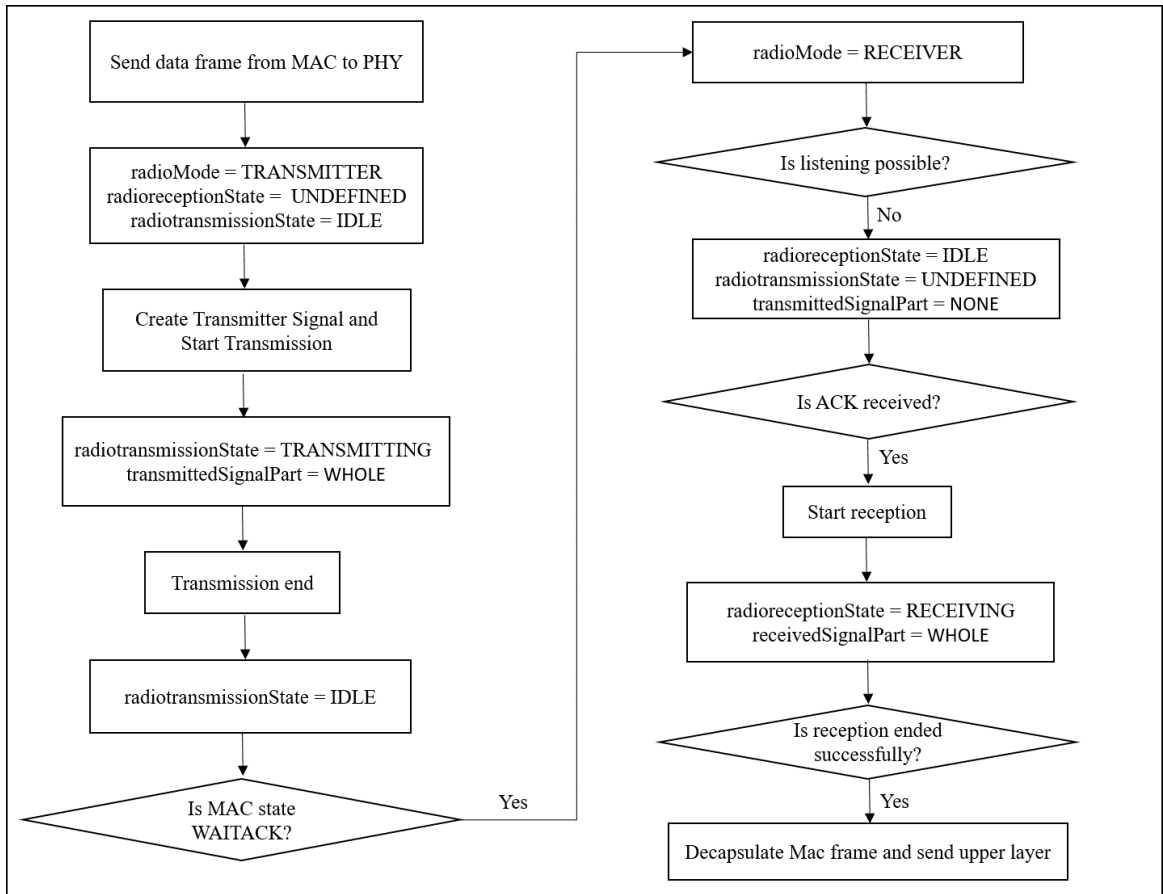


Figure 3.13. PHY layer coordination with MAC layer

## 4. CRN MODEL UNDER TWO CHANNEL

The two-channel CRN has been designed inspired from crSimulator. crSimulator is a discrete simulation model developed in Omnet++ for cognitive radio adhoc networks. Together with the modules it offers, it provides the infrastructure to improve multiple channel CRN design. Therefore, it has been preferred to develop the two-channel CRN design based on this study.

### 4.1. PU HOST ARCHITECTURE

Protocol stack of PU is kept simple in crSimulator. A single layer has been implemented for PU as shown in Figure 4.1 and it is ensured that only packets are produced at this layer such as the APP layer and then packets are transmitted to the medium.

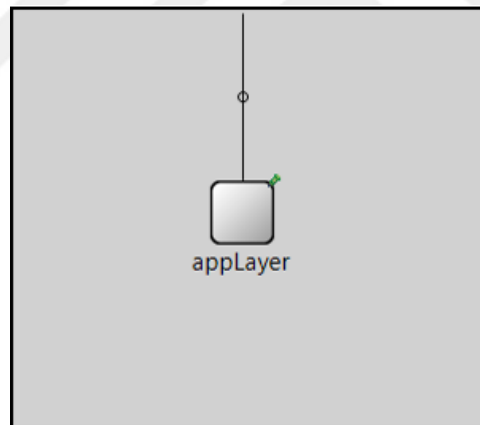


Figure 4.1. PU host structure in crSimulator

### 4.2. MODIFIED PU HOST ARCHITECTURE

In crSimulator, the PU node is kept simple and it is only intended to occupy the medium by transmitting. In this way, SUs are enabled to perceive the medium as empty or busy for transmission. In that study, it is not focused on the transmission of PUs. The transmitter PU does not have recipient PU. In this thesis, the transmission between PUs in CRN has been taken into account and it is aimed to have a more realistic communication system for PUs. For this reason, the PU protocol stack in crSimulator has been developed to create a real

network and the fully developed protocol stack for PU nodes has been implemented. In addition, receiver PU node is designed to make real PU communication. Since CRN has the transceiver PU, communication information and successful packet transmission for PUs have gained importance.

APP, Transport, Network, MAC and PHY layers have been implemented for PU nodes as shown in Figure 4.2. While the transport layer is only responsible for transmitting data and control messages, the network layer provides host to select destination address. APP and MAC layers specific to PUs have been developed. The PHY layer models the transceiver. Broadcasting is done by adding transmission information to the packets.

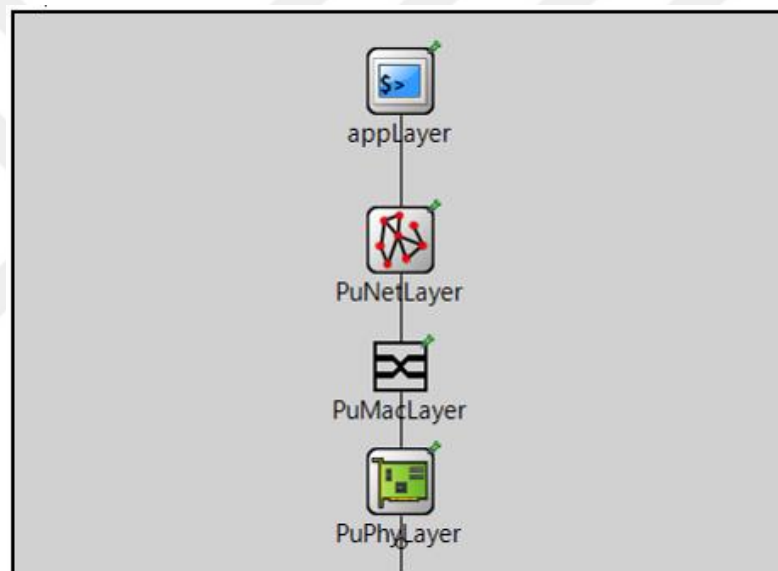


Figure 4.2. Modified model for PU host

#### 4.2.1. APP Layer for Modified PU

In crSimulator, PU node only has APP layer. In this layer, the PU performs data transmission at regular intervals and keeps the environment busy. The arrival interval of the PU is determined with the idle duration. Since there is only PU transmitter, a timer called busy duration is kept in order to notify the SUs that the PU transmission is over. Busy duration determines how long the PU makes the channel busy. During this time, the SU detects the channel with the PU transmission as busy. At the end of busy duration, PU transmits from this layer a message called the PUEND, which indicates that the transmission of PU is

finished. When SUs receive this transmitted message in the spectrum sensor module, they realize that the channel is free and set it as free.

The APP layer of the modified PU node contains only the arrival rate to define the transmission start time of the selected transmitter PU. The busy duration is not used because the receiver PU node is also in the CRN. Keeping the channel busy is handled by the MAC layer. After receiving the data packet from the transmitter PU, the receiver PU sends ACK using the acknowledgment mechanism. When the data-ACK messaging is complete between PU pair after the transmitter PU receives the ACK message, a timer is set to keep the medium busy at MAC layer. After this timer expires, the PU transmitter transmits the end of the transmission (EOT) message indicating that the PU transmission has ended. SU can understand that the PU transmission is over when the EOT from the PU is received by the spectrum sensor of SU.

Also, channel assignment is performed for each PU at this layer. It is determined which channel is used by which user.

#### **4.2.2. MAC Layer for Modified PU**

The MAC layer for PUs is designed based on IEEE 802.11b protocol. Acknowledge mechanism and retransmission perform on PUs. Acknowledge mechanism is used in order to understand whether the packet transmitted by the transmitter PU reaches the receiver PU successfully or not. For this, an ACK timer is kept for the transmitter PU. After receiving the data, the receiver PU is provided to generate and send ACK packet at the MAC layer. If the ACK packet is not received on the transmitter side within the ACK timer, the frame is sent again. Developed MAC layer for transmitter PU is shown in Figure 4.3.

In this way, the transmission success for PU becomes questionable. In cognitive radio, it is the main purpose for SUs to use the medium without affecting the transmission of PUs. SUs must use the channel in the absence of PU. In order to prove this in the two channel CRN model, it is necessary to perform PU transmission. It provides to verify whether PU transmission is affected by SUs in two channel CRN by defining MAC protocol for PUs.

Moreover, real communication is provided for PUs due to acknowledge mechanism in PU. Since PU data-ACK communication in MAC layer takes place within the same second, the

timer is kept in order to keep the channel a little more busy after the transmitter PU receives ACK. At the end of this timer, an EOT message is generated at the MAC layer and transmitted to the environment. This message informs that the PU communication is finished and notifies the SUs that the channels are no longer used by the PUs.

In this way, SUs can now detect that the transmission of PUs is over and the channel is empty after the data - ACK messaging between PU pair instead of the busy duration indicator from APP layer in crSimulator.

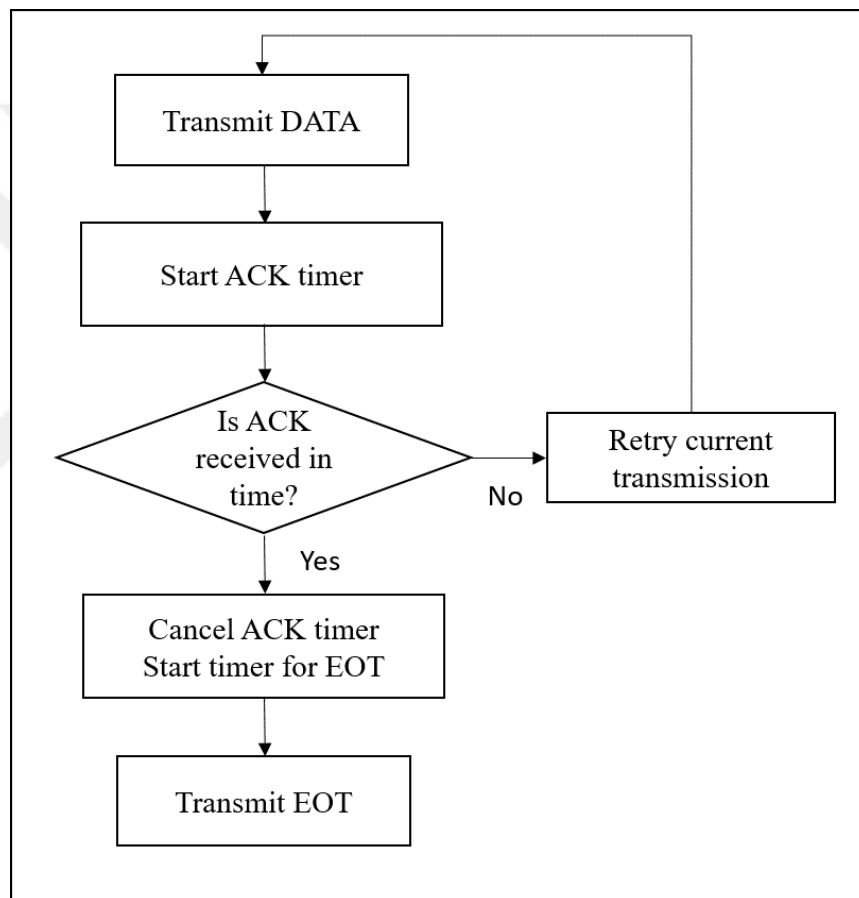


Figure 4.3. Modified MAC layer for transmitter PU

### 4.3. SU HOST ARCHITECTURE

The SU node consists of a complete protocol stack and coordinated modules.

### 4.3.1. Layers of Protocol Stack

In crSimulator, SU host is presented as a fully developed protocol stack. Figure 4.4 shows SU host structure. SU host has APP, Transport, Network, MAC and PHY layer.

The APP layer is responsible for generating packets for transmission at certain intervals. Network and transport layers are kept simple.

The MAC layer is based on the IEEE 802.11b standard. Acknowledge is provided in the Link layer. Since more than one channel is defined in this study, channel mobility is dynamic, so access to the environment is important. SU detects the presence of PU by periodically sensing channels. It detects empty and busy channels, then it selects the empty channel and performs the channel switching. MAC protocol parameters can be set in the .ini configuration file.

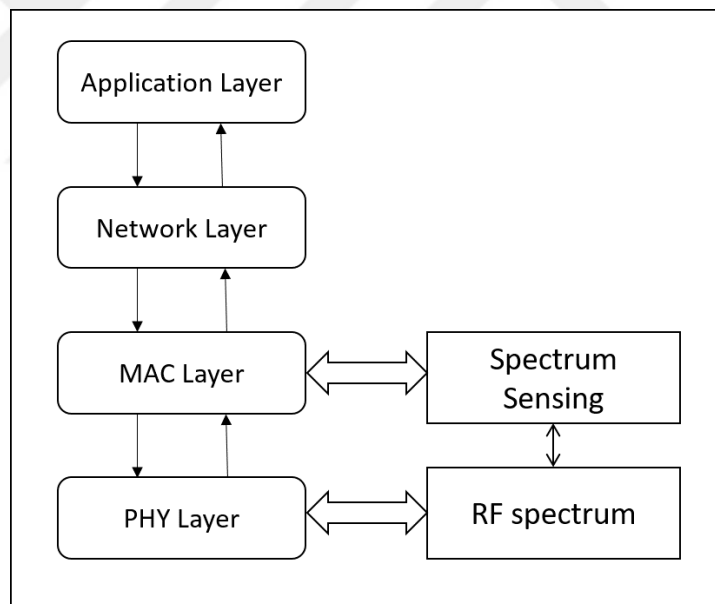


Figure 4.4. SU host structure

### 4.3.2. MAC Layer of SU

The MAC protocol based on IEEE 802.11b is implemented in the MAC layer for SUs. RTS/CTS mechanism is preferred as DCF method.

According to this protocol, after the channel is detected as empty, the transmitter SU node sends RTS on this channel and starts the RTS timer as shown in Figure 4.5. If the channel is detected as empty after receiving RTS, the receiver SU responds with CTS. If the CTS packet is received by the transmitter SU within the RTS period, the RTS timer is cleared and data transmission starts.

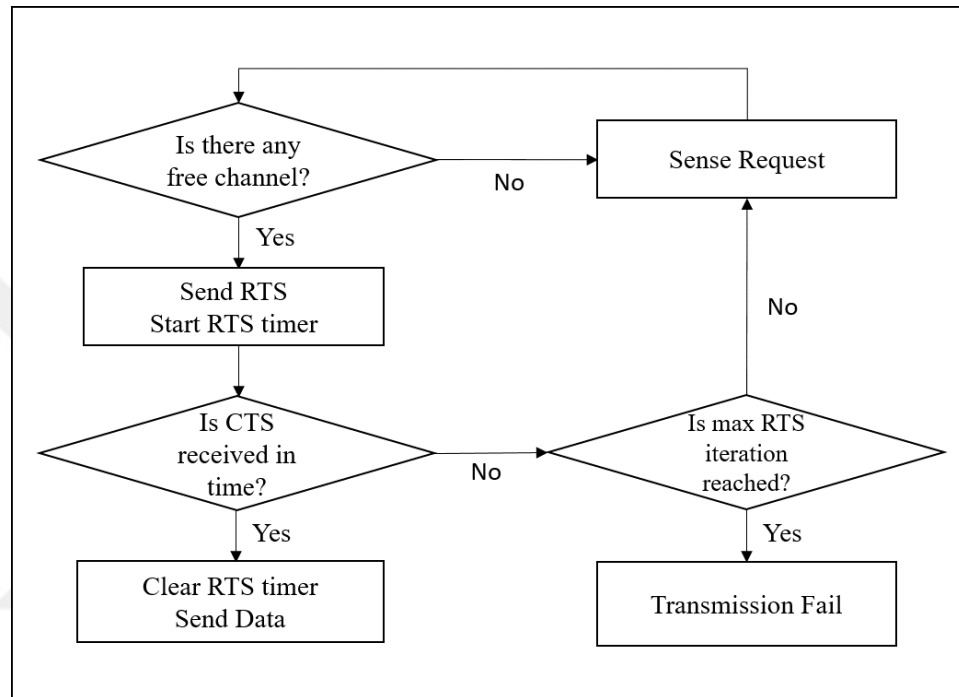


Figure 4.5. RTS /CTS mechanism at MAC layer of transmitter SU

If the receiving SU cannot find a free channel to transmit CTS after receiving the RTS, it responds with negative acknowledgment (NACK) as shown in Figure 4.6. NACK is not sent over any channel, so the channel is not set.

If the NACK packet is received within the RTS period at the transmitter SU, RTS timer is cleared. Then, if the empty channel is sensed, the RTS is sent again.

The number of RTS iteration can be set in the .ini file. If CTS is not received when the maximum number of RTS iteration is exceeded, transmission fails for that packet and a new packet is generated.

MAC layer works integrated with the spectrum sensor. It receives the information from the spectrum sensor whether the channels are empty or not.

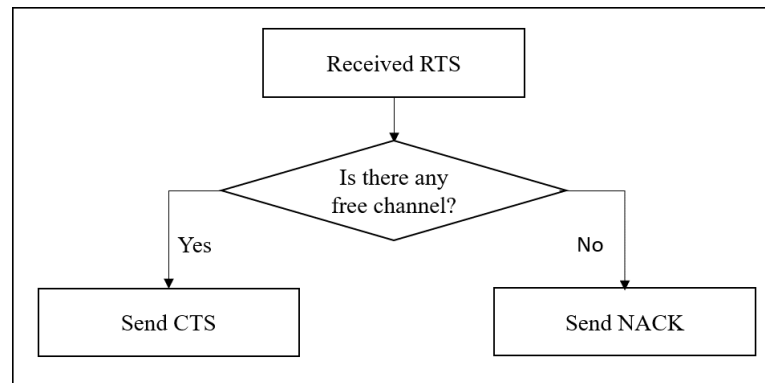


Figure 4.6. RTS /CTS mechanism at MAC layer of receiver SU

After the receiver SU receives data, it responds to the transmitter SU by sending an ACK through the proposed channel if it is still free. If the current channel is not free, it responds with NACK. If the ACK reaches to the transmitter SU, the ACK timer is cleared and the data is sent by starting the ACK timer again at transmitter side. If the NACK reaches to the transmitter SU, the ACK timer is cleared and waits until the empty channel is detected. When the empty channel is detected, the RTS/CTS mechanism takes place again and the data-ACK flow is started after the RTS/CTS transmission is successfully completed.

### 4.3.3. Modules for SU Host

In the crSimulator, additional modules have been developed to perform the cognitive radio functions for SUs apart from the protocol stack.

#### 4.3.3.1. Spectrum Sensor Module

The spectrum sensor module is designed to sense channels for SUs. The module detects the presence of PU in the channels. Spectrum sensor detects when PUs start transmitting and sets the channel to which the user is transmitting as busy. Then, when transmission of PU finishes, it sets the channel as free.

Channels are detected at certain intervals and it keeps the information of whether the channel is empty or busy for each channel. In this way, available channels are found for SUs to transmit.

This module has interface with each layer. It works in collaboration with the MAC layer and the PHY layer. The MAC layer requests the free channel information from the spectrum sensor in order to use the channels. The spectrum sensor module reports the free channel information to the MAC layer with this request. In addition, every packet coming from outside is delivered to the spectrum sensor from the PHY layer and the spectrum sensor sets the channel as free or busy according to this message. Thus, SUs have the opportunity to communicate by detecting the free channel.

Spectrum sensor handles requests from MAC layer. If all channels are detected as busy at MAC layer, a request to sense for the empty channel is sent from the MAC layer of both the receiver and the transmitter SU to the spectrum sensor.

Before sending CTS, MAC layer of the receiver SU sends a request to the spectrum sensor to find whether the channel on which it receives RTS is still free.

Every message received from the outside is transmitted to the spectrum sensor from the PHY layer. If the received packet is a packet addressed to itself, if it is a data or CTS packet, the current data channel is set as the channel from which this packet comes.

When it receives a message that is not addressed to itself, the spectrum sensor acts according to the type of message. If it receives a data packet that is not addressed to it, it sets the channel which the packet comes as busy. Also, if data packet comes from PU, this is reported to the MAC layer from spectrum sensor.

If it receives an ACK message that is not addressed to it, it is realized that the PU transmission is over and makes the channel that ACK comes free.

#### ***4.3.3.2. RF Spectrum Module***

The module modifies the packet according to the transmission parameters by looking at the control information of the packets. This module determines the characteristics of the channel such as data rate.

#### **4.3.4. Modified Spectrum Sensor Module**

The transmitter node is defined for the PU in crSimulator. PU communication is carried out with a single node. Since there is no receiver PU in the crSimulator, after the transmitter PU

has started its transmission, it initiates a timer which is called busy duration to notify the SUs that the transmission has ended. During this time, the SUs perceive the channel PU is transmitting as busy. At the end of busy duration, a message is sent from the transmitter PU to notify the SUs that the transmission is finished. After this message comes to the spectrum sensor of the SUs from the PHY layer, the channel from which the message comes is considered empty.

Since the PU also has a receiver node in the CRN we designed, the PU pair makes a complete transmission with data/ACK messaging. After this transmission is completed, the channel is kept busy for a while and then the EOT message is transmitted indicating the end of the PU transmission. When the spectrum sensor receives this message, it realizes that the data-ACK messaging between PUs is finished so the channel is empty.

#### **4.3.5. MAC Layer with Spectrum Sensor**

The communication of the SUs takes place by the cooperation of the spectrum sensor and the MAC layer as seen in Figure 4.7. Free channel detection and transmission of RTS/CTS and data/ACK packets are realized with this cooperation. Cooperation works as follows: When a data packet comes to the MAC layer of transmitter SU from the APP layer, the MAC layer sends the sense request to the spectrum sensor. With this request, empty channels are expected to be determined. Sensing the channels in the spectrum sensor takes place randomly. After the channels are sensed by spectrum sensor, the result is reported to the MAC layer. If a free channel is detected, the RTS message from the MAC layer is sent over this channel. After RTS reaches the receiver SU, it inquires whether the proposed channel is still empty by sending a sense request to the spectrum sensor. The spectrum sensor looks if the queried channel is free or not and returns this information to the MAC layer. If the proposed channel is still free, CTS message is transmitted from MAC layer of receiver SU. However, if the proposed channel is not empty, a NACK message is sent. When the NACK message is sent, no channel is set.

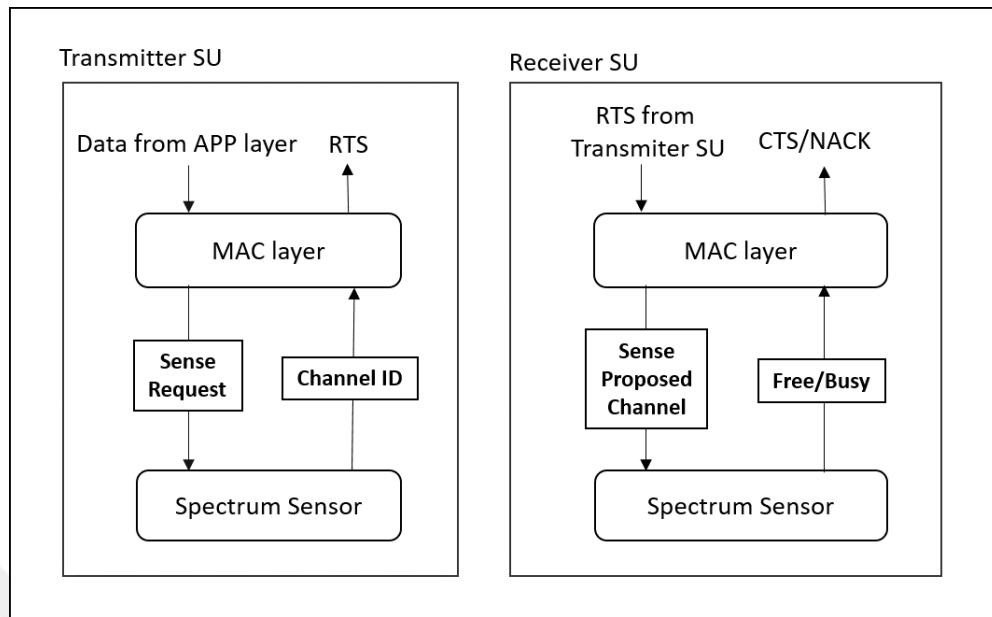


Figure 4.7. Cooperation of MAC layer and spectrum sensor

CTS message comes to spectrum sensor from PHY layer at transmitter SU. All messages received from the PHY layer are transmitted to both MAC and spectrum sensor. At spectrum sensor, the current data channel is set as the channel where the CTS comes from. When CTS comes to the transmitter SU at MAC layer, it checks whether the proposed channel is still free. If it is still free, RTS timer is cleared and data is sent. Then, ACK timer is scheduled. If proposed channel is not free, a sense request is sent to the spectrum sensor again and the RTS timer is cleared. After, data comes to the receiver SU, ACK packet is sent. The transmitter SU node that receives ACK clears the ACK timer and sends the next data after it.

When the PU starts transmitting while the SU is transmitting, the data packet sent by the PU reaches the PHY layer of the SU. Then, it is sent to the spectrum sensor from PHY layer and the channel from which this data packet comes is set as busy at spectrum sensor module. This information is reported to the MAC layer. At the MAC layer, if the node is transmitting and waiting for ACK, the ACK timer is cleared. Sense request is sent to the spectrum sensor module. The spectrum sensor reports the free channel information to the MAC layer. According to this information, RTS is sent from MAC layer if there is a free channel. Otherwise, it sends sense request for free channel information. If the SU node is in receiving state when PU starts transmission, NACK is sent. After transmitter SU receives the NACK,

it sends a message to the spectrum sensor to get information if there are other free channels. According to the response, RTS and data are sent on other free channel, respectively.

Meanwhile, when the receiver PU sends an ACK in its own transmission and then, transmitter PU notifies the end of its transmission with the EOT message. The SU node evaluates it in the spectrum sensor and sets the received EOT message channel as free.

#### 4.4. CRN MODEL IN CRSIMULATOR

The CRN created in crSimulator consists of one PU and one receiver - transmitter SU pair shown in Figure 4.8. The PU is used as a transmitter. It is only used to generate packets at certain intervals and keeps the environment busy. There is no destination addressed as PU receiver for receiving PU packets. Therefore, there is no point-to-point transmission for PU and it does not have an acknowledge mechanism that detects whether the addressed packet has arrived or not. For SUs, receiver and transmitter nodes providing point-to-point transmission are used.

Two data channels are implemented for transmission. The SUs enable to transmit over the free channel.

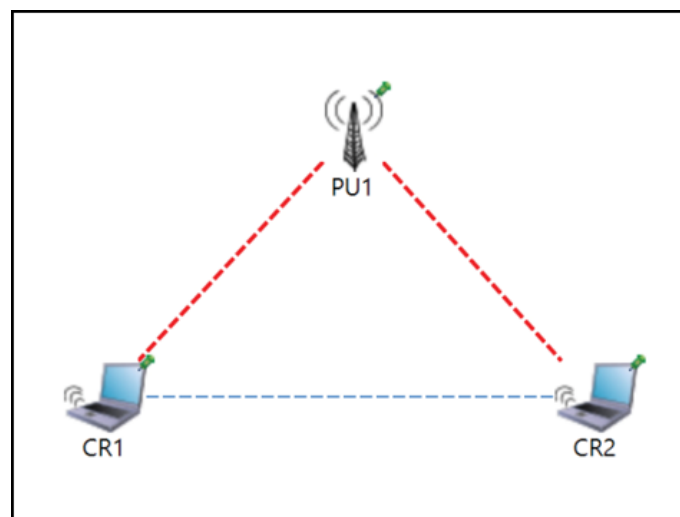


Figure 4.8. CRN model in crSimulator

#### 4.5. DEVELOPED CRN MODEL WITH TWO CHANNEL

It is necessary to model new CRN that has transmitter – receiver PU pair with MAC layer due to the shortcomings of the crSimulator mentioned in the previous part.

In our one-channel CRN design, it is not possible for both PU and SU to transmit at the same time, since there is only one channel where the PUs and SUs transmit. SUs can only utilize the medium while the PUs are not transmitting. To enable transmission of PUs and SUs at the same time, two-channel CRN design is realized. In our two-channel CRN design, there are two spectrum bands to be shared by users. Thus, SUs can transmit simultaneously with PUs on different channels from PUs if one of the two channels is not used by PU. It is sufficient for the SU communication to find a single channel empty.

The CRN simulation model consists of two PU transceiver pairs and one SU pair in Figure 4.9. Each PU pair has its own private channel for communication. When PUs do not communicate, SUs can utilize the channel belonging to that PU pair.

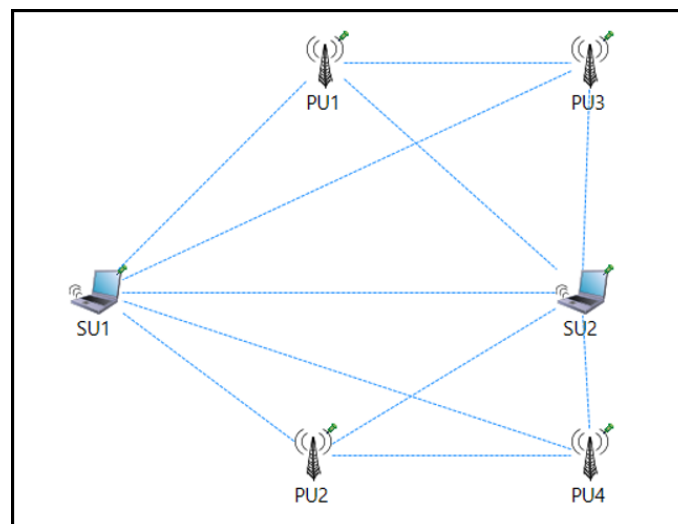


Figure 4.9. CRN model with two channel

## 5. RESULTS AND DISCUSSIONS

In this chapter, we introduce both CRN models. We describe the simulations for each CRN and evaluate the their obtained results.

### 5.1. CRN MODEL UNDER SINGLE CHANNEL RESULTS

In CRN simulation model, PUs and SUs share the same spectrum. In CRN model, we consider two PUs, acting as a transmitter and a receiver, and two SUs, acting as a transmitter and a receiver. In this case, PUs and SUs are within the communication and interference range of each other. In Figure 5.1, HostC and HostD represent the PU transmitter and the PU receiver respectively, while HostA and HostB represent the SU transmitter and the SU receiver respectively.

While the destination address and port number of HostC show HostD, the destination address and port number of HostA indicate to HostB. The designed PU module is used for HostC and HostD as the MAC protocol, while the CsmacaMac module in INET is used for HostA and HostB. For all users, the UnitDiskRadio module in INET is defined for the PHY layer.

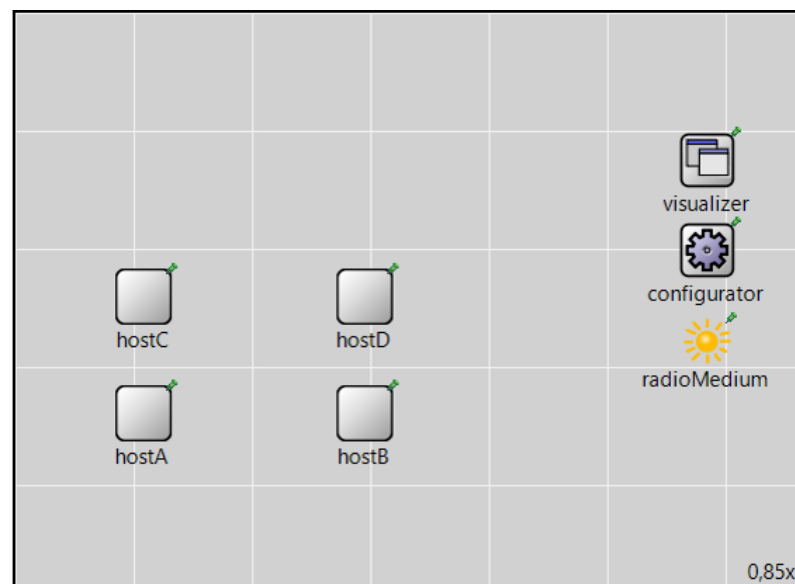


Figure 5.1. CRN model under single channel

### 5.1.1. Simulation Parameters

The simulation time, communication and interference range, ACK timeout and bit rate values are kept constant in the CRN simulation. The bit rate of both PU and SU transmitter-receiver pairs is set as 1 Mbps. The ACK timeout and the overall simulation duration are selected as 300  $\mu$ s and 20 seconds, respectively.

Packet size and packet send interval parameters of the hosts are varied to analyze the number of transmitted and received packets for each user type. The sendInterval parameter defines which user starts the first transmission. Packets are sent at sendInterval intervals, where sendInterval is set as an exponential random variable. Then, the packet size is set by the messageLength parameter. Packet sizes are considered 200, 500 and 1000 bytes for both PU and SU transmission. Constant parameters used in the single channel simulation are shown in Table 5.1.

Table 5.1. Parameters in the single channel simulation

<b>Constant Parameter</b>	<b>Value</b>
Simulation time	20 sec
ackTimeout	300 $\mu$ s
Bit rate	1 Mbps
sendInterval	exponential(6ms) exponential(12ms)
messageLength	200, 500, 1000 bytes
Frequency	2.4 GHz

### 5.1.2. Simulation Scenarios

In the CRN model with a single channel, different simulations are obtained to verify designed CRN model by changing the packet size and packet send interval.

Two scenarios have been investigated in the CRN with one channel model. In the first case, the PU and the SU hosts are in each other's communication and interference range 250 meters and 500 meters respectively. In the second case, relay is used to enable transmission between the SUs since the SU transmitter and the SU receiver are not in each other's communication range. Send interval and message length are two input parameters in both scenarios.

#### 5.1.2.1. First Scenario

The first case consists of the PU and SU transmitter-receiver hosts which are in each other's communication range.

In this scenario, firstly, two different packet sending intervals are set for the transmitter users in the CRN. The effect of different packet sending intervals on packet transmission is investigated by keeping the packet size constant.

Then, different packet sizes are set in each packet sending intervals. Thus, the effect of packet size on PUs and SUs in terms of number of transmitted and received packets is examined.

First, the parameter `sendInterval` is set as exponential (6ms) for the SU transmitter and exponential (12ms) for the PU transmitter. In this case, the SU starts transmission before the PU. Since the PU transmits less frequently than SU, this case gives the SU more opportunity to find the medium idle.

In this simulation, SU starts transmitting first. At this point, transmission time for PU comes and packet is generated from APP layer then, PU starts transmission. Since the SU transmission is not finished, a collision occurs with the transmitting data or ACK of the SU. After the collision, the PU immediately starts the transmission of the colliding packet without waiting for a certain time. On the other hand, SU retransmits the colliding packet only when the medium is detected as empty, as it has to listen medium according to the DIFS and backoff time.

Later, the parameter `sendInterval` settings have been reversed as exponential (6ms) for PU transmitter and exponential (12ms) for SU transmitter. In this case, PU sends the first packet to the medium. Because the packet transmission range of PUs is less, they use the medium more frequently, which reduces the SU's opportunity to use the medium. SUs find the opportunity to transmit when they find the environment empty. However, while SU is transmitting, PU may start transmitting and collision may occur.

All simulations are performed by changing the packet sizes to 200, 500 and 1000 bytes for each user.

#### ***5.1.2.2. First Scenario Results***

The results obtained when `sendInterval` parameter is set to exponential (6ms) for SU transmitter and exponential (12ms) for PU transmitter are shown in Figure 5.2. In this case, the SU starts transmission before the PU. The PU transmits less frequently than SU, so the SU has more opportunity to find the medium idle.

Figure 5.2 contains results for different cases. Firstly, SUs are excluded from the simulation, and only PUs are simulated with the same parameters. In other words, HostC transmits to HostD where HostA and HostB are not present. Then, the number of transmitted and received packets of PUs is obtained and results are shown in Figure 5.2.

Additionally, Figure 5.2 shows the effect of packet size on the number of transmitted and received packets for both PU and SU when SU first utilizes the medium in the CRN.

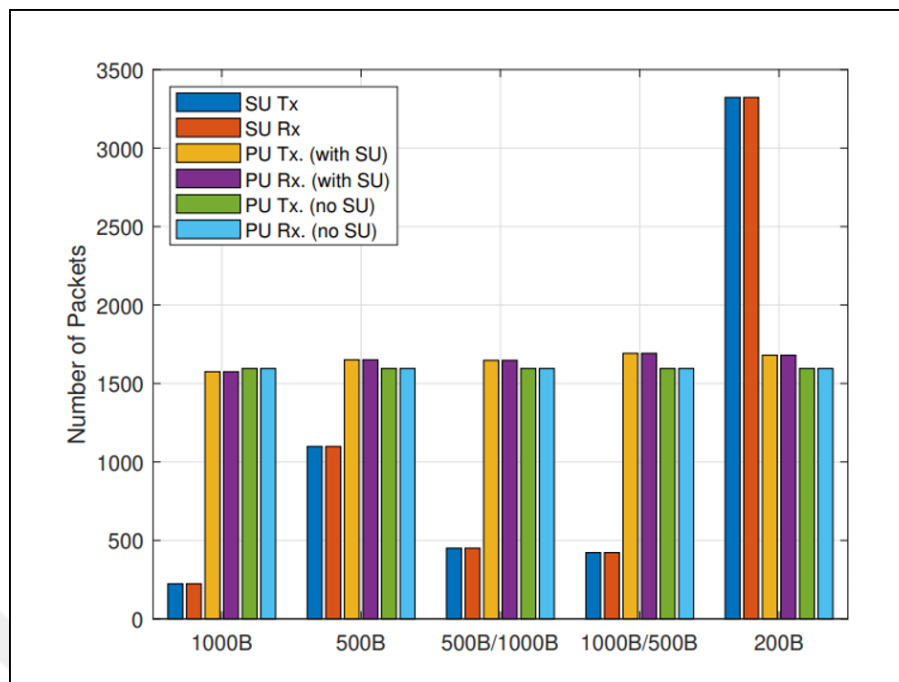


Figure 5.2. Number of transmitted and received packets with SU average send interval 6ms, PU average send interval 12ms

Simulation results show that the presence of SU transmission does not affect the number of transmitted and received PU packets. PU transmits and receives same number of packets under both presence and absence of the SU transmission.

It is also shown that the packets of both PU and SU can be transmitted without packet loss between the transmitter and receiver through the retransmission of the collided packets. The average number of PU transmitted and received packets for each packet size is quite close to each other. All packets generated by the APP layer have been transmitted.

For the SU transmitter-receiver pair, it is observed that as the packet size decreases, the number of SU packets transmitted and received increases. This is due to the fact that, as the SU packet size decreases, the SU medium occupation time also decreases. Furthermore, the increase SU activity (as the packet size decreases) does not degrade the number of transmitted and received PU packets.

Then, parameter `sendInterval` settings have been reversed as `exponential(6ms)` for PU transmitter and `exponential(12ms)` for SU transmitter as seen Figure 5.3. In this case, PU sends the first packet to the medium. This figure shows, like the Figure 5.2, results obtained

when only PUs are present in the simulation. In addition, Figure 5.3 shows the impact of PU and SU transmission on the number of transmitted and received packets for CRN with given transmission intervals and variable packet size.

In this case, when there are no SUs in the simulation, the number of packets transmitted and received by the PUs is almost the same as the results obtained in the CRN when SUs are added in the simulation. The presence of SUs in the CRN does not affect PUs.

PU transmits packets more frequently than SU to the medium because PU transmission interval is shorter than the SU. Hence, the SU finds less opportunity to transmit its packet. For example, when the packet size is set as 1000B, it seems that the SU can never sense the medium idle so there is no SU transmission. However, as the packet size decreases, SU starts to perform packet transmission. When the packet size is set as the smallest 200B, the number of packets transmitted increases to 1500 packets.

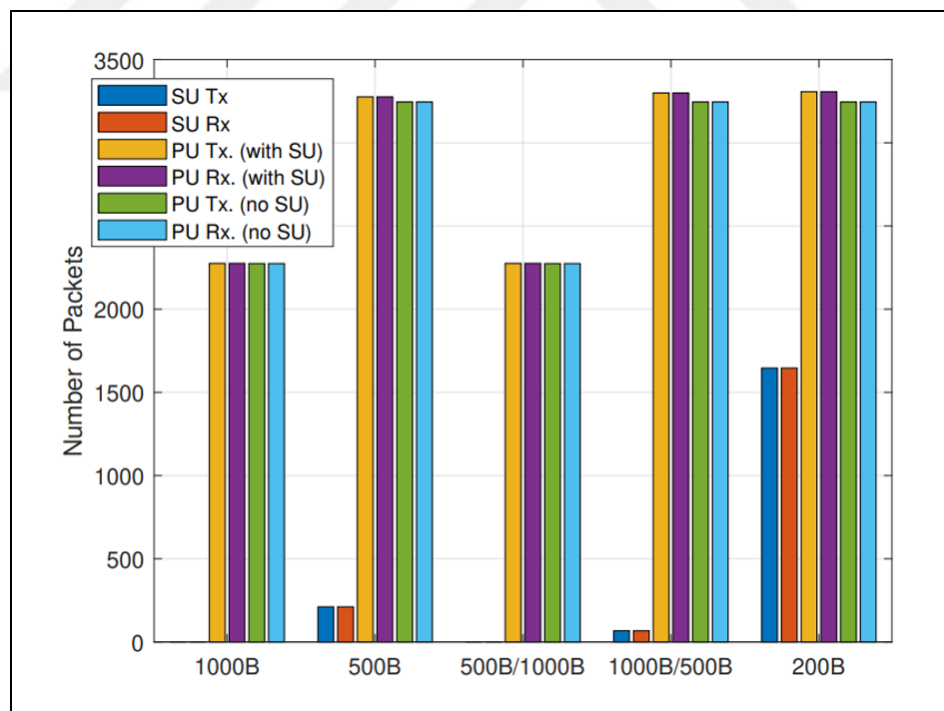


Figure 5.3. Number of transmitted and received packets with SU average send interval 12 ms, PU average send interval 6 ms

When it is further compared Figure 5.2 and Figure 5.3, it is observed that the number of packets transmitted by PU is doubled in Figure 5.3 with respect to Figure 5.2. As the smaller transmission interval of PU causes higher number of PU transmission in the CRN. Reducing the sending interval of PU results in less packet transmission of SU as compared to the first test result.

Furthermore, when the packet size of PU is varied, the number of transmitted and received packets increases as the PU packet size decreases. For the case presented in Figure 5.3, PU packets are generated at the APP layer on the average in every 6ms. Note that these packets are first stored in the MAC queue before transmission, and then sent to the PU receiver. As a particular instance, over a 20 seconds simulation duration, when we use PU packet sizes of 200B and 1000B, the APP layer generates 3311 and 3284 packets respectively. On the other hand, since the 200B packets are transmitted fast as compared to 1000B packets, the receiver captures 3306 and 2275 packets respectively for 200B and 1000B packet sizes. Then, the difference  $3311-3306=5$  and  $3284-2275=1009$  represent the number of packets left in the MAC queue when 200B and 1000B packet sizes are respectively used.

#### **5.1.2.3. Second Scenario**

In this scenario, it is considered that the SUs in the CRN are not in each other's communication range, a relay is used to provide transmission between the SUs. As shown in Figure 5.4, the SU receiver HostB is outside the communication range of the SU transmitter HostA, and hostR1 acts as a relay is placed in the communication range of both.

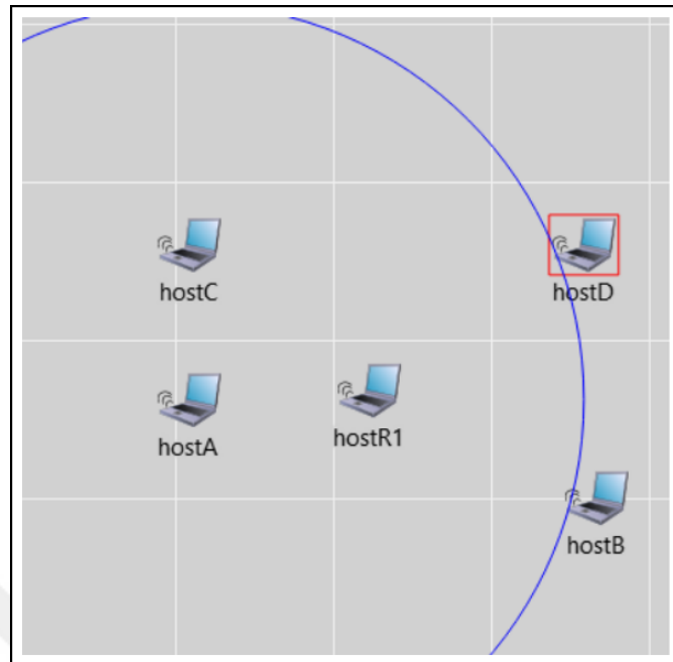


Figure 5.4. One channel CRN model with SU relay node

In relay scenario, packet send interval is changed with varying packet size parameters. The number of transmitted and received PU/SU packets is examined when the sendInterval parameter is set as exponential (6ms) for the SU transmitter and exponential (12ms) for the PU transmitter. Moreover, the number of transmitted and received PU/SU packets is analyzed when the sendInterval parameter is set as exponential (12ms) for the SU transmitter and exponential (6ms) for the PU transmitter.

CsmaCaMac module is used for HostR1 so HostR1 is also SU. It forwards the packet received from HostA to HostB. It transmits the ACK received from HostB to HostA. Due to the SU characteristic, HostR1 transmits the packets by listening to the environment. Therefore, it takes a long time to transmit data and ACK from one end to the other end for successful transmission. However, as a relay, it ensures the successful end-to-end transmission of the packet.

#### **5.1.2.4. Second Scenario Results**

In this scenario, we consider the SUs in the CRN are not in each other's communication range, and we use a relay to provide transmission between the SUs. The SU receiver is outside the communication range of the SU transmitter and hostR1 acts as a relay is placed in the communication range of both. Figure 5.5 shows the number of transmitted and

received PU/SU packets when the sendInterval parameter is set as exponential (6ms) for the SU transmitter and exponential (12ms) for the PU transmitter. Then, Figure 5.6 shows the number of transmitted and received PU/SU packets when the sendInterval parameter is set as exponential (12ms) for the SU transmitter and exponential (6ms) for the PU transmitter.

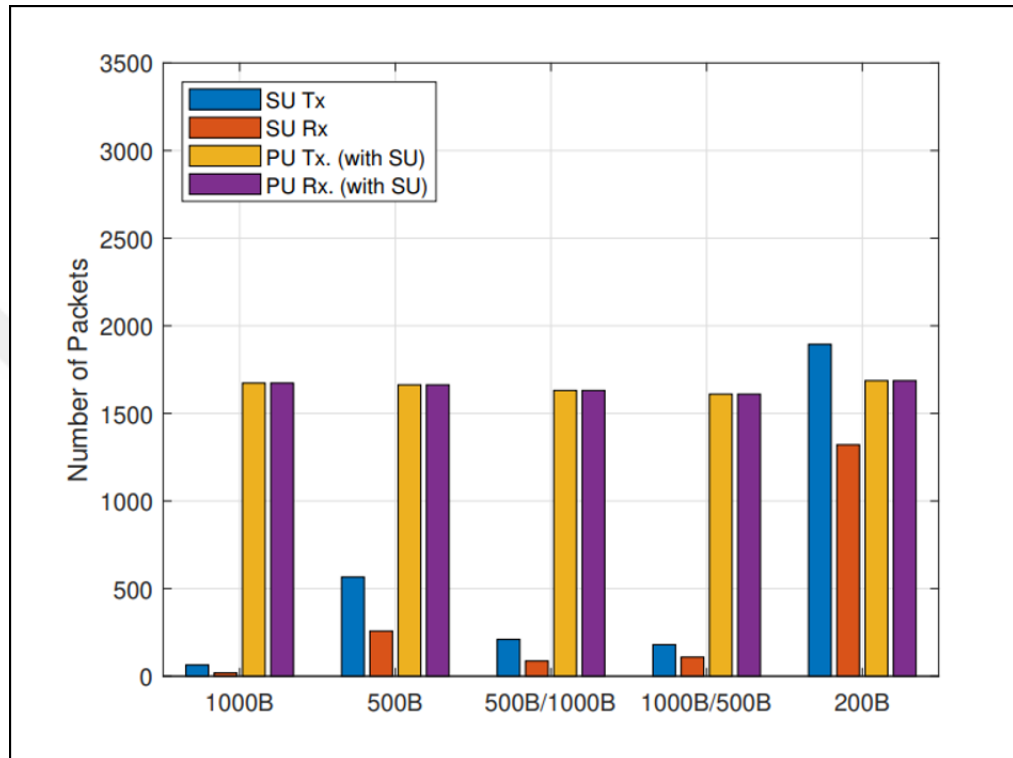


Figure 5.5. Number of transmitted and received packets SU average send interval 6 ms, PU average send interval 12 ms, with relay

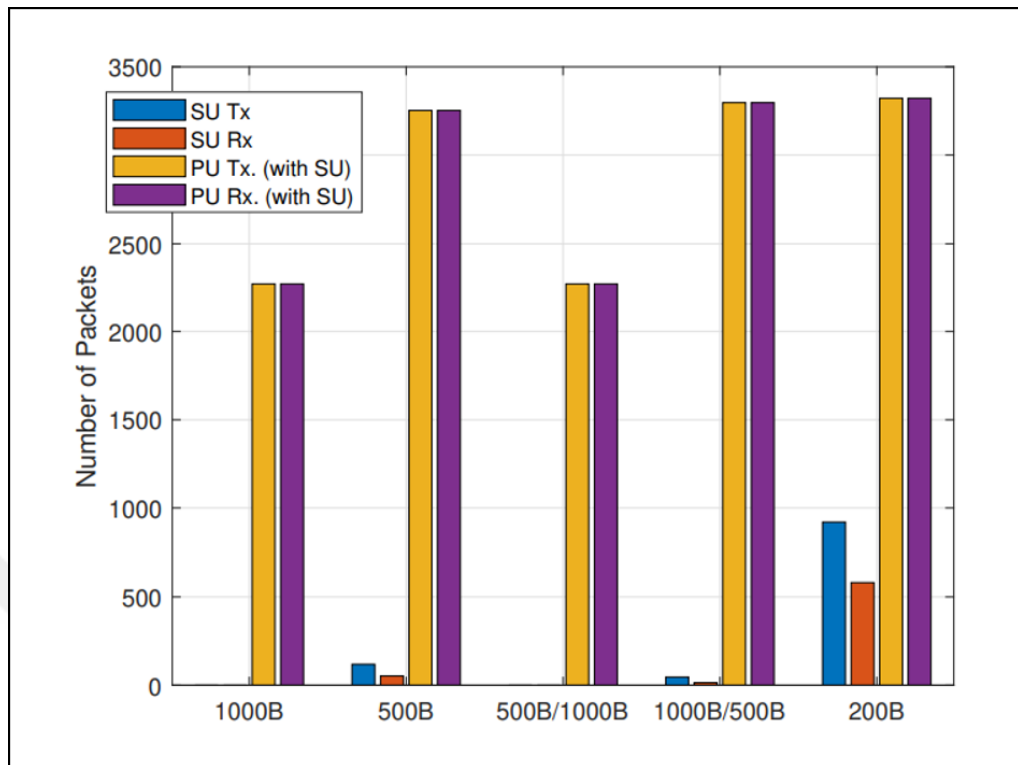


Figure 5.6. Number of transmitted and received packets SU average send interval 12 ms, PU average send interval 6 ms, with relay

When we compare Figure 5.5 and Figure 5.6 with Figure 5.2 and Figure 5.3, we observe that the number of PU transmitted and received packets is almost the same for each packet size, which means SU relaying does not affect the number of transmitted and received PU packets.

Furthermore, as shown in Figure 5.5, where the send interval of SU is 6ms, there is almost no SU transmission at all packet sizes other than the packet size of 200 and 500 bytes. A significant number of packet is transmitted at 200 bytes. When the send interval of SU is increased by 12ms in Figure 5.6, the SU is able to transmit at a packet size of 200 bytes, as the medium is more occupied by the PUs. Therefore, in both cases, SU has the chance to transmit in the smallest packet size.

In case of a relay, it is observed that there is packet loss between the SU transmitter and the SU receiver. This is due to the fact that it takes a longer time for the packet to reach the receiver SU from the transmitter SU and return ACK from the receiver SU via the relay. During this time, the PU may enter the medium and may cause a collision. Then, the SU does not have the opportunity to enter the medium to complete its transmissions. This is also

verified in Figure 5.6, where the increased PU activity (6ms) as compared to the SU activity (12ms), significantly decreases the number of transmitted and received SU packets.

## 5.2. CRN MODEL UNDER TWO CHANNEL RESULTS

In the two-channel CRN simulation, there are PU pairs PU1/PU3 and PU2/PU4 as shown in the Figure 4.9. PU1 and PU2 are transmitter PUs while PU3 and PU4 are receiver PUs. A channel is assigned to each pair. Each PU couple has their own channel that they use. The SU1 and SU2 transceivers are SUs.

### 5.2.1. Simulation Parameters

In the two-channel CRN simulation, the simulation time is 20 seconds. In all simulations, the time that the SU first transmits to the environment is set to 0.01 seconds. In other words, the first packet for the transmitter SU comes from the APP layer at this time. The results are obtained by setting the size of the data packets 500 and 1000 bytes.

The acktimer is set to 0.01s. For PUs, after data-ACK messaging is completed, it is waited 0.05s before sending an EOT message.

The idle duration, which is the arrival interval of PUs, is changed. Two channel CRN simulation parameters are shown in Table 5.2.

Table 5.2. Parameters in the two channel simulation

Constant Parameter	Value
Simulation time	20s
ackTimer	0.01s
EOT timer	0.05s
messageLength	500 /1000 Bytes
Data rate	2.04Mbps

### 5.2.2. Simulation Scenarios

With the developed two-channel CRN model, different scenarios are created. It is aimed to show that SUs can use the channels that belong to the licensed PUs when the license owners are not using them and then, the SUs leave the channels when the license owners start using the channels again.

With the developed two channel CRN model, impact of the SU on the PU is shown by taking different simulations. At the same time, with two PU pairs and two channels, the transmission success of the SU and the availability of the channels are examined according to the transmission frequency of the PUs. By changing the channel usage intervals of the PUs, the effect on the transmission of the PUs and SUs is investigated.

In addition, it is observed how the SU transmission is affected if the PUs simultaneously utilize their channels or if one PU is transmitting on one channel and the other channel is empty. Moreover, the effect of the presence of SU in the CRN environment on PU transmission is examined. Apart from these, the transmission of SUs is carried out through a relay acting as a SU and its effect on the transmission of users is investigated with different simulations.

#### 5.2.2.1. First Scenario

In the first scenario, two different kinds of PU idle duration are set as constant intervals and exponential random values. Idle durations of PUs are set to 0.1s, 0.3s, 0.5s, 0.7s, 0.9s as constants and exponential random values in five different cases. Five different simulations are taken according to each PU idle duration calculation method. For each simulation, the packet sizes of PUs and SUs are set 500 and 1000 bytes. The idle durations of both transmitter PUs are changed at the same time. First transmission time of the SU is fixed and 0.01s. In order to verify that the CRN created in the simulations works in accordance with the working principles of the cognitive radio, the successful transmission numbers of PUs and SUs are examined.

In addition, the parameters are kept the same and simulations are taken when there are only two PUs pair while there is no SU pair user in the CRN. Thus, it is investigated whether the SU affects the transmission of the PU by participating in the CRN. In other words, it is aimed to show that the SU does not affect the PU transmission in CRN.

Moreover, the packet sizes of PUs and SUs are doubled and the simulation results are obtained so the effect of packet size on the transmission is investigated.

In case the idle duration of both transmitter PUs is kept the same and constant value, PUs transmit at the same time with idle duration, so both channels are kept busy. Both channels become free except idle duration of PUs. Thus, when the transmitter SU wants to occupy the channel, either both channels are empty and it is able to transmit packets, or it is not able to transmit because both channels are busy. In this scenario, the SU starts transmitting first. Since the PUs have the same time to begin communication, both start transmitting at the same time. Each PU transmits on its assigned channel. When the data packet transmitted from the PUs comes to the spectrum sensor of the SU, the spectrum sensor detects both channels as busy and reports it to the MAC layer. The MAC layer sends a request to the spectrum sensor for the free channel. The MAC layer does not transmit until one channel is detected as empty by the spectrum sensor. When the both receiver PUs send the ACK packets, the spectrum sensor of the SU receives these ACK packets and sets the both channels from which the packets come to free. Then, spectrum sensor reports it to the MAC layer. Thus, the SU can use the channel. In this scenario, channels are simultaneously occupied and left free by PUs.

When the idle durations of the PUs are set exponentially, there are cases where both channels are busy at the same time or one channel is busy and the other channel is free.

While SU is communicating on a channel, the PU that owns the channel can start the transmission on that channel. In this case, the SU spectrum sensor detects that the channel is busy from the data packet coming from the PU and starts to sense the other empty channels. When it finds a free channel, it continues its transmission.

If the PU starts its transmission on a channel other than the channel used by the SU, SU continues to transmit over its own utilized channel without being affected by the PU transmission.

#### **5.2.2.2. First Scenario Results**

Two different situations are investigated in this scenario. As a variable parameter, the idle duration of the both PUs has been increased by 0.2 from 0.1 to 0.9 seconds as constant and exponential value at the same time. Figure 5.7 shows the simulation results obtained by setting the idle duration of PUs as constant, while Figure 5.8 shows the case where the idle

duration is set with exponential value. In each idle duration, the number of PU transmission is investigated when there is no SU in the simulation and there is a SU pair in the CRN. The simulations are carried out by setting the packet sizes of PUs and SUs to 500 bytes and 1000 bytes, while SUs are in the CRN medium and not.

As can be seen in the Figure 5.7 and Figure 5.8, the transmission number of the PUs does not change with or without the SUs and PU transmission is not affected by the change of packet size because PU is set to transmit only once at each idle duration.

During the simulation period of 20 seconds, the number of PU transmission is roughly obtained from equation in (5.1) when idle duration is set constant in Figure 5.7.

$$\text{number of PU transmission} = \frac{\text{simulation time}}{\text{idle duration of PU}} \quad (5.1)$$

When idle duration is set as the exponential parameter in Figure 5.8, the average of the total transmission numbers of both PUs corresponds to equation in (5.2).

$$\text{average number of PU transmission} = \frac{\text{simulation time}}{\text{idle duration of PU}} \quad (5.2)$$

Both Figure 5.7 and 5.8 show that the transmission of SU does not affect the transmission of PUs because the SU only transmits when it finds an empty channel. That is, if the channel is not used by the PUs, the SU transmits. With this result, it has been seen that this concept, which is the main idea of the cognitive user, is realized in our model.

Figure 5.7 and 5.8 also show the effect of the idle duration of PUs on the transmission of PUs. Both the figures show that the shorter the idle duration of the PUs, the more transmissions they have.

In addition, it is shown that the effect of the idle duration of PUs on the SU transmission. The longer idle duration of the PUs, the more transmissions of SU because as PUs utilize channels less, SUs sense the channel as empty more and transmit. In Figure 5.7, since both PUs transmit at the same time, SUs can find both channels empty and transmit, except for idle duration moments.

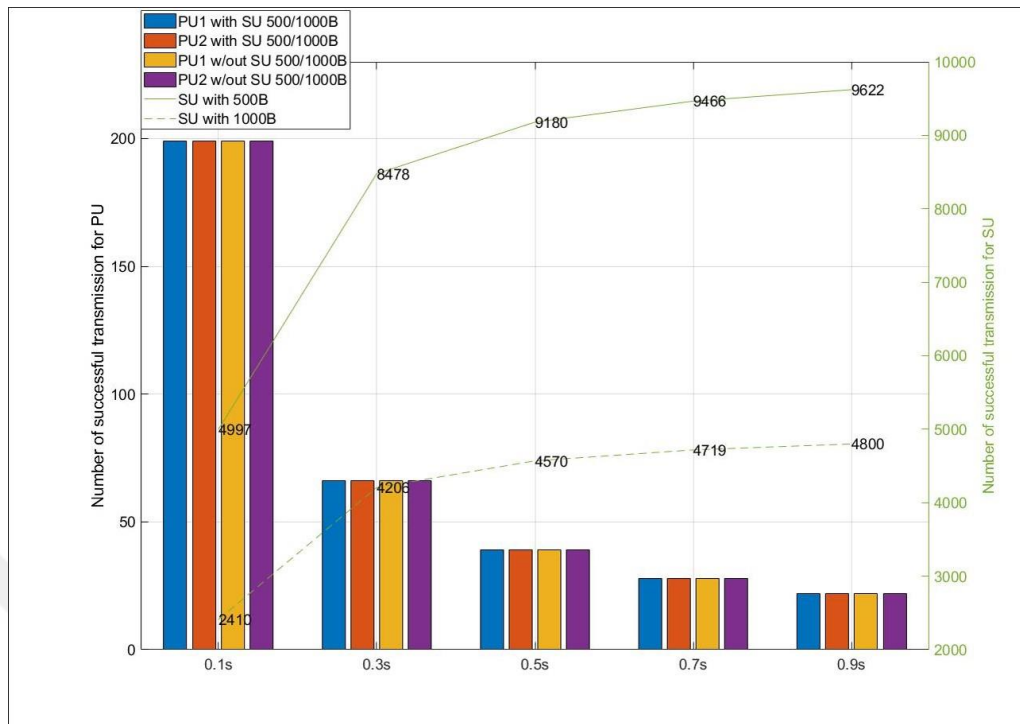


Figure 5.7. Number of PU and SU successful transmission with constant idle duration

In Figure 5.7, number of successful transmission for SU reaches approximately 5000 packets at 1000 bytes packet size when idle duration of PUs increases to 0.9 seconds. Under this condition, data/ACK transmission for one packet is completed in 0.004 seconds. If SU transmits its packet when PU does not transmit during simulation time, maximum number of successful transmission for SUs is approximately 5000 packets from equation (5.3).

$$\text{number of SU transmission} = \frac{\text{simulation time}}{\text{transmission time of a packet}} \quad (5.3)$$

In Figure 5.8, both channels can be busy or empty at the same time or a single channel may be busy. If PU starts transmission on a different channel than SU while SU is transmitting, this does not affect SU transmission. However, if PU transmission starts on the same channel with the SU transmission, the SU must switch the channel and this takes from the SU transmission time and causes the transmission number to decrease.

In Figure 5.7 and Figure 5.8, the number of packets transmitted by SUs in 500 bytes is twice the number of packets transmitted in 1000 bytes. As the packet size increases, less transmission takes place in the same time period because it takes more time to transmit the packet.

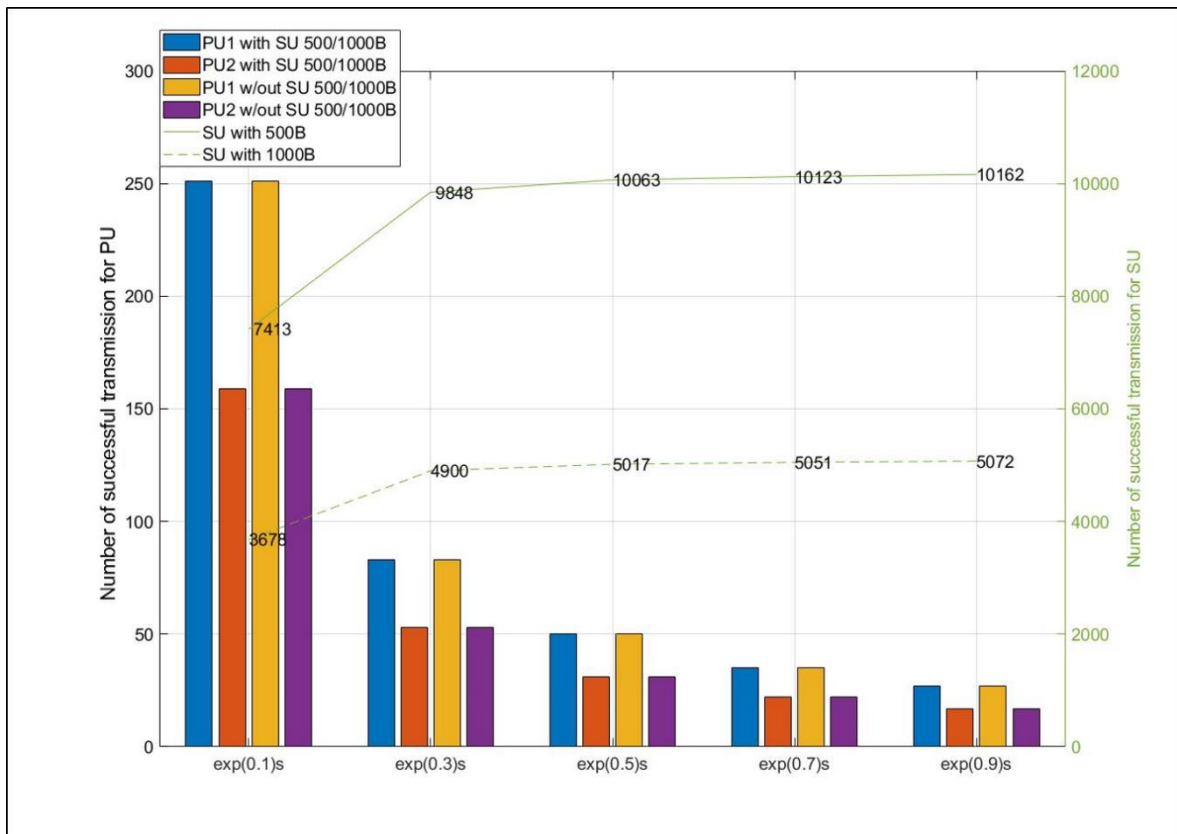


Figure 5.8. Number of PU and SU successful transmission with constant exponential idle duration

### 5.2.2.3. Second Scenario

In this simulation, the idle duration of one PU is kept constant as 0.1s, while the idle duration of the other PU is increased to 0.3s, 0.5s, 0.7s, 0.9s. Idle durations are set as constant and exponential value. With these values, different simulations are taken.

In case the PU idle duration is set to constant value, while there are moments when only one channel is kept busy by one PU during the simulation, there are moments when both channels are busy by both PUs. In other words, while the PU with 0.1s idle duration is in transmission, the other PU is also in communication. Therefore, when the SU wants to transmit, sometimes it can find a channel empty and transmit, but it cannot transmit when both channels are busy by PUs.

In the case where PU idle duration is set as exponential value, while one PU is kept busy the medium at constant time intervals, the medium occupancy time of the other PU is changed.

In both cases, if PU the licensed owner of the channel, starts to transmit while SU is transmitting on the same channel, the spectrum sensor sets the channel as busy. Then, it checks if there is other free channel and notifies the other unused channel to the MAC layer. In this way, SU can continue to transmit on the free channel. Both PU and SU can continue to communicate over different channels at the same time.

In addition, when SU is transmitting, if PU starts transmitting on the different channel, the SU is not affected by this transmission and can continue to transmit on the channel it uses. While the spectrum sensor sets the used channel as busy, since the channel used by the SU is still sensed as empty, the SU is not affected by transmission of PU. When the transmission of PU on the channel is finished, the spectrum sensor sets this channel as empty again.

As a result, the effect of the exposure time of the PU, whose idle duration is variable, on both PU and SU transmissions is examined. The simulations are performed by setting the packet size 500 and 1000 bytes for each case. In simulation, the effect of SUs in CRN on PU transmission is examined.

#### **5.2.2.4. Second Scenario Results**

In this simulation, the idle duration of one PU is kept constant and the idle duration of the other one is changed. While PU transmission takes place on one channel SU can select other free channel and transmit there.

Figure 5.9 shows the results when the idle duration is constant value and the Figure 5.10 shows the results when it is exponential value.

As can be seen from the Figure 5.9, the number of transmission for PU with the constant idle duration (0.1s) is the same in each simulation. It is the ratio of the simulation duration divided by the idle duration ( $20s / 0.1s$ ). In Figure 5.10, transmission number of PU, whose transmission range is kept constant, is the same in all cases. Therefore, the other PU entering or leaving the environment with varying idle duration or the presence of the SU does not affect the transmission of the PU whose idle duration is kept constant.

In addition, it is shown that the number of PU transmission does not change when the simulation is taken in the absence of SU in both figures.

In Figure 5.9, it is observed that the number of PU transmission is decreased with the increasing idle duration of the PU whose idle duration changed, in the ratio of the simulation

duration divided by the idle duration. In Figure 5.10, where the idle duration is changed exponentially, it is observed that the transmission of the PU, whose idle duration is changed, increased with decreasing PU idle duration.

There is a change in the transmission number of the SU with the change of the PU transmission. The PU with variable idle duration affects the number of SU transmission. With the increasing idle duration of the PU, SU senses the channel more empty so it find opportunity to communicate. As a result, both figures 5.9 and 5.10 show that the number of SU transmission increases with increasing idle duration of PU.

When effect of packet size on the transmission is examined, it can be seen that small packet size causes more packets to be transmitted for the transmission of SUs, while PU transmission is unaffected for both Figures 5.9 and 5.10 because PUs are only allowed to transmit one times at the moment of idle duration. When the packet size is increased from 500 bytes to 1000 bytes, the number of packet transmission is halved for SUs.

When Figure 5.7 and 5.9 are compared, it can be said that the number of SU transmission is close to each other. While both channels are kept busy by PUs at the same time in Figure 5.7, one channel or both channels may be busy in Figure 5.9.

For example, in Figure 5.7, when the idle duration is 0.3s for both PUs, the PUs simultaneously occupy both channels during the simulation period (20s) at moments that are multiples of 0.3s. Here, both channels are kept busy by PUs for approximately 66 times which is simulation time (20s) divided by idle duration (0.3s).

In Figure 5.9, where one PU has idle duration of 0.1s and the other PU has idle duration of 0.3s, both channels are kept busy by PUs at multiples of 0.3 second. Apart from that, one channel is kept busy at multiples of 0.1s by one PU, which has idle duration of 0.1s. Here, both channels are kept busy at the same time approximately 66 times which is simulation time (20s) divided by idle duration (0.3s). Moreover, a single channel is kept busy by the PU with idle duration of 0.1s approximately 140 times.

While the transmitted SU packet in Figure 5.7 is 8478, it is 8447 in Figure 5.9. In the case when a single channel is busy 140 times, if PU starts transmission on the same channel with SU, this slows down the SU transmission as SU must switch channel. Then, it leads to decrease in SU transmission. However, if PU transmission starts on the channel where the

SU does not transmit, this does not affect the SU transmission. It is expected that the number of transmitted SU packets is slightly higher than the previous situation so this explains why the number of packets transmitted by SU is not much different from each other in Figure 5.7 and 5.9.

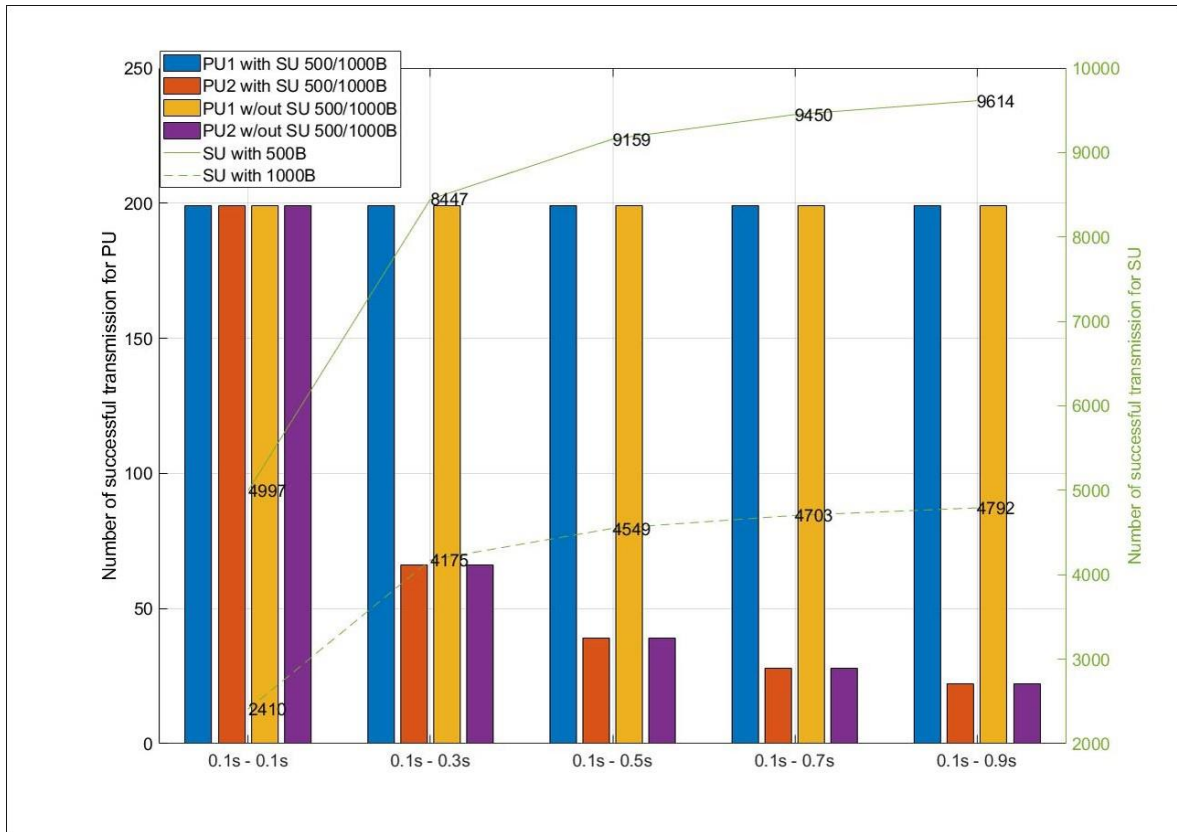


Figure 5.9. Number of PU and SU successful transmission with varying idle duration

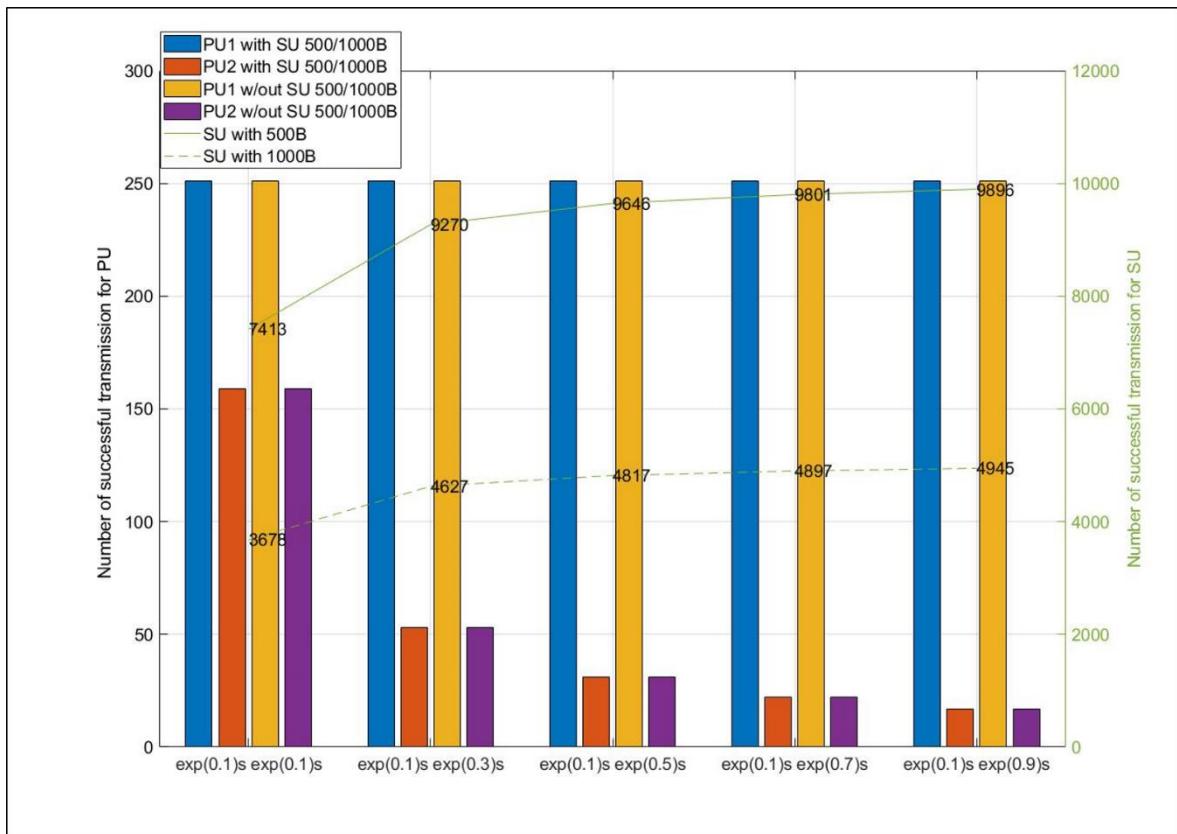


Figure 5.10. Number of PU and SU successful transmission with varying exponential idle duration

### 5.2.2.5. Third Scenario

In this scenario, it is intended that SUs communicate with each other through a relay that acts as a SU. In Figure 5.11, a SU3 acting as a relay is placed between the transceiver pair SU1 and SU2. SU1 and SU2 communicate over SU3. SU3 forwards the data packet from SU1 to SU2 and the ACK packet received from SU2 to SU1.

SU3 used as a relay listens to the environment before starting the transmission. It makes RTS/CTS messaging with SU1 and SU2. If the channel is empty, it transmits. It does not transmit if the channels are busy.

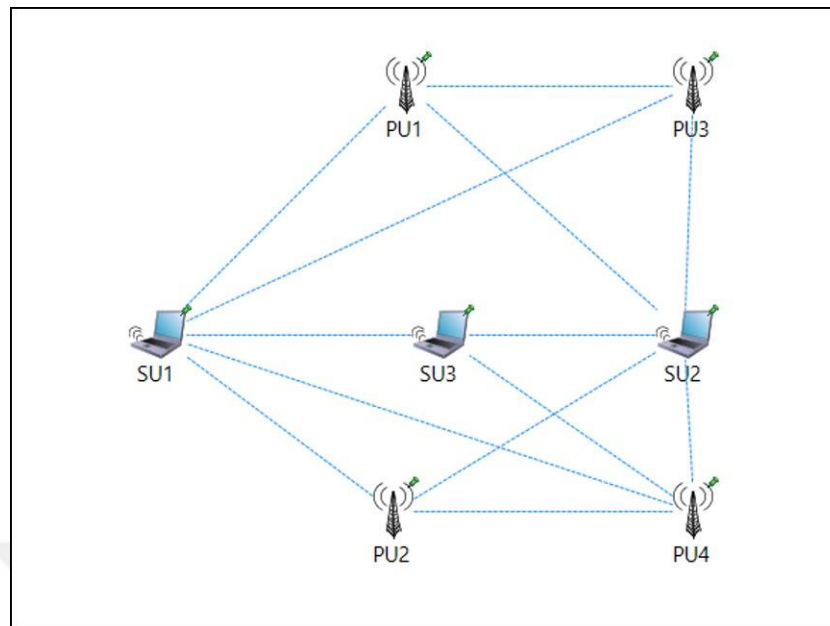


Figure 5.11. Two channel CRN model with SU relay node

Firstly, the idle durations of the both PUs are set to 0.1s, 0.3s, 0.5s, 0.7s and 0.9s at the same time and five simulations are taken. The packet size in all simulations is 500 bytes. Both PUs start transmitting at the same time and occupy the medium at the same time so both channels are free and busy at the same time.

SU1 starts its first transmission in 0.01s. RTS/CTS messaging takes place between SU1 - SU3 and between SU3 - SU2. If the channels are detected as empty, SU1 starts transmitting data over the channel determined by the spectrum sensor. SU3 receives data packet and forwards this packet to SU2. When SU2 receives data, it responds with ACK. After SU3 receives the ACK packet, it forwards ACK to SU1. If SU1 receives ACK within the ACK timer, SU1 starts transmitting the next data packet. When PU transmission starts while SU communication continues, spectrum sensors of all SUs set the channels as busy. Then, SU2 and SU3 send NACK. SU1 receives NACK from SU3 and SU3 receives NACK from SU2 and they send requests from their MAC layer to spectrum sensor in order to sense the free channel. Spectrum sensor senses the environment until it finds an empty channel. When channels are free, it reports the empty channel to the MAC layer and transmission continues after RTS/CTS messaging for SUs.

As the second case in the relay scenario, the idle duration of one PU is kept constant while the idle duration of the other PU is increased from 0.1s to 0.9s in 0.2s increments. With these values, five simulations are taken with the packet size set to 500 bytes.

In this case there are moments when both channels are occupied by both PUs at the same time and one channel is occupied by one PU.

In moments when a single channel is occupied by a PU, if the PU starts transmitting on the channel where the SU transmits, this channel is perceived as busy by the spectrum sensor of the SUs. With the start of PU transmission, SU2 and SU3 send NACK. A request is sent from MAC layer of SU1 and SU3 to spectrum sensor to sense empty channel. Spectrum sensor detects the other free channel and gives empty channel information to the MAC layer. Thus, while a channel is occupied by the PU, SU can continue to transmit over the empty channel.

However, while the SU is transmitting, if the PU starts its transmission over the channel where there is no SU transmission, the SU continues its transmission over the channel and transmits without being affected by the presence of the PU.

#### **5.2.2.6. Third Scenario Results**

Figure 5.12 shows the simulation results where the idle durations of the both PUs are set to the same value at the same time in the relay scenario. The results of five different simulations are shown. The idle durations of both PUs are set to 0.1s, 0.3s, 0.5s, 0.7s, 0.9s.

It is observed that as the idle duration of the PUs increases, the number of PU transmission decreases.

In Figure 5.12, it is seen that the number of successful transmission for the SU pair in the case of relay increases as the idle duration of PU increases. As the idle duration of PU increases, the SU pair has more opportunity to transmit because the medium is detected as empty for a longer period of time.

The results of the SUs shown in Figure 5.7, in which the SU pair communicates without a relay with the same parameters, are also included in Figure 5.12 for comparison purpose. When two results are compared, it is observed that the number of SU successful transmission is halved when the SU pair (SU1- SU2) communicates through the relay (SU3) compared to the case where the SU pair communicates without a relay under the same conditions. In the

relay case, the PU may delay the completion of SU transmission or cause the retransmission due to the fact that the PU starts transmission before the data/ACK messaging of the SU is finished. It also takes longer time for packets to be transmitted over the relay compared to transmitting the packet directly to the receiver without a relay. Therefore, a reduction in successful transmission of SU is observed in the case of relay.

Moreover, the SUs do not affect the PU transmission in the relay condition.

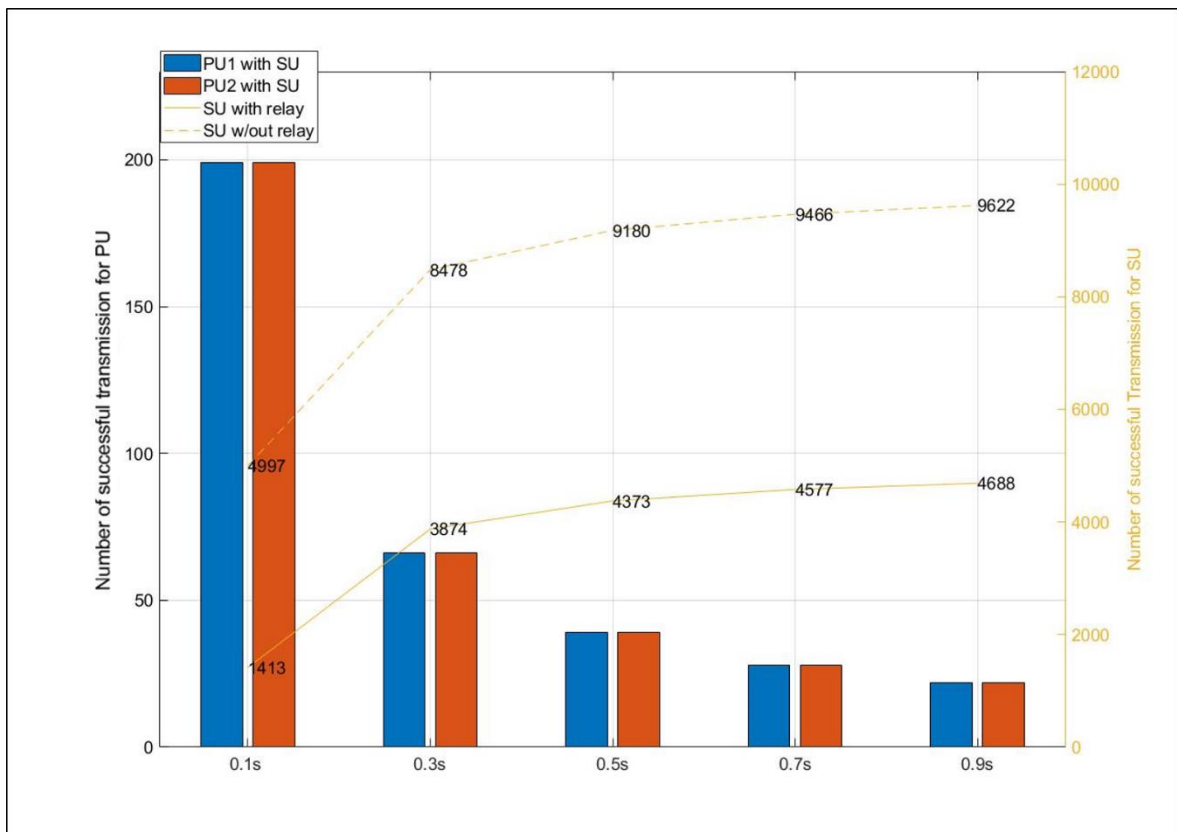


Figure 5.12. Number of successful transmission for PU and SU with constant idle duration under relay condition

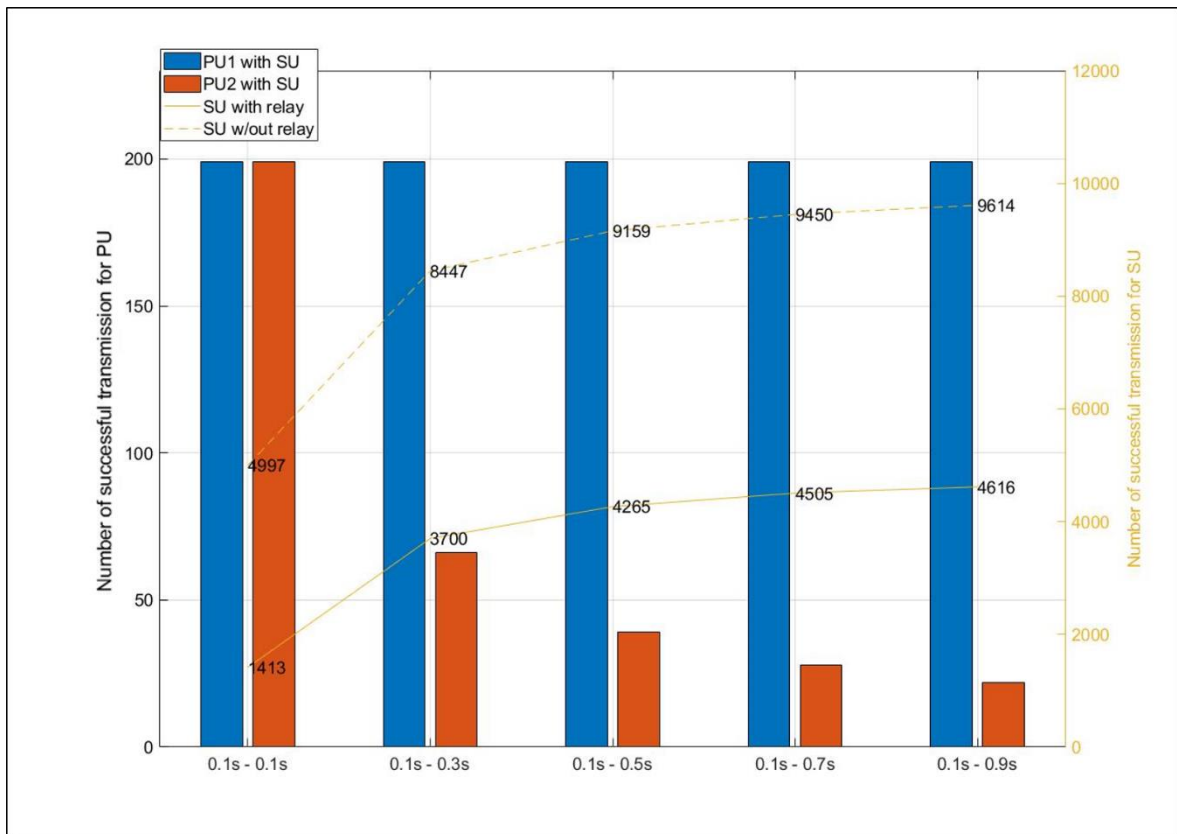


Figure 5.13. Number of successful transmission for PU and SU with varying idle duration under relay condition

Figure 5.13 shows the results when the idle duration of one PU is kept constant at 0.1s and the idle duration of the other PU is increased in 0.2 increments up to 0.9s. It is seen that the successful transmission number of the PU, whose idle duration is kept constant in each simulation, is equal to the simulation duration (20s) divided by idle duration (0.1s). On the other hand, it is seen that the transmission number of the PU with variable idle duration is equal to the simulation duration (20s) divided by idle duration(s) and its transmission decreases with increasing idle duration of PU. This also proves that the presence of SU does not affect the PU transmission due to the fact that the PU transmission is equal to the simulation time divided by idle duration.

An increase in SU transmission is observed with increasing idle duration of PU, whose idle duration is variable.

In Figure 5.13, the simulation result shown in Figure 5.9 for SUs, which is performed without a relay under the same conditions, is given. Here, it is seen that the number of successful transmission for the SU in the relay case is halved compared to the non-relay case.

In addition, when the transmissions of SUs are compared in Figure 5.12 and 5.13, the number of transmission in the same packet size is almost same.

For example, in Figure 5.12, when idle durations of both PUs are 0.5s, PUs simultaneously occupy the both channels 40 times for simulation time (20s) divided by idle duration (0.5s). In this case, it is seen that SUs successfully transmit 4373 times.

In Figure 5.13, when the both channels are occupied by one PU every 0.1s and other PU every 0.5s, both channels are simultaneously busy at times that are multiples of 0.5s. This equals to approximately 40 times which is simulation time (20s) divided by idle duration (0.5s). Other than that, a single channel is occupied approximately 160 times. In this case, SU successfully transmits 4265 times. When a single channel is occupied by one PU, if the SU is not transmitting on that channel, the SU is not affected by this transmission. However, if the PU starts transmitting over the channel on which the SU transmits, this causes the SU to switch channel. Thus, this reduces the number of transmission for SU in the same time period. This explains why there is slightly less successful transmission in the case where 4265 transmissions are made compared to the previously described situation, which has 4373 successful transmission for SU.

## 6. CONCLUSION AND FUTURE WORK

In this thesis, we modeled an interweave single channel and two channel CRN in Omnet++. The single-channel model was made use of INET Framework and two-channel model was created based on the work of the crSimulator, which offers a library to create multi-channel CRN. In both models, PU design was developed based on the shortcomings in INET and crSimulator. Receiver and transmitter PU designs were carried out in two modelled CRNs. These nodes have been developed in accordance with PU transmission in the cognitive radio concept. PU nodes contain acknowledge and retransmission mechanism with fully protocol stack. Thus, a complete communication system has been provided for the PUs in CRN. A transceiver PU has been developed, which can transmit regardless of whether the channels are empty or busy, without sensing the environment.

In the single channel model, users communicate over a single channel. For the transmission of SU, the transmission of PU must not be on the channel. PU can start its transmission while SU is in transmission. In this case, a collision occurs. After this collision, the PU is first allowed to utilize the medium.

Later, two-channel model was created where PUs could operate on different frequency channels so that SUs have more opportunities to transmit. SU senses the entire spectrum with spectrum sensor and determines the idle bands before transmission. The sensing mechanism of the spectrum sensor has been modified to detect the start and end of the PU transmission in the channel. While SU is transmitting, it is given the opportunity to switch channel so that it can transmit over the channel not occupied by PU. Since there are two channels in this CRN model, both PU and SU can transmit at the same time.

We presented the effectiveness of our model under varying PU and SU packet sizes, under varying PU and SU packet send interval times, and under the case where the transmission between the SU transmitter and the SU receiver is performed over a relay node.

For the one channel CRN, our method ensured that PU received what it transmitted and SU had the best performance with a small packet size under a relatively long PU send interval time. It has been observed that the colliding packets were transmitted without packet loss by retransmission for both users. In case of a relay, the relay node also checked the PU

availability. Therefore, the destination node received less packets than transmitted from the source node.

For the two channel CRN, SUs achieved the best efficiency with the highest PU idle duration and smaller packet size when both channels were used by PUs simultaneously. In the case where the idle duration of one PU was kept constant and the varying idle duration of the other PU, SU showed the best performance with the highest idle duration of PU. In the relay case, the number of SU successful transmission decreased, but PU transmission was unaffected.

The model we created was verified by taking various simulations. The effect of the presence of SU on PU transmission was examined on both CRN models, and it was shown that PU transmission was not affected by the presence of SU. This shows that the SU, which is the basis of CRN, is able to use the channels without affecting the PU transmission in our model.

The increase in the frequency of PU medium occupation caused the SU to transmit less. This showed that in CRNs, opportunity to access the environment for SUs depends on PU activity.

Packet collision is not defined in crSimulator because crSimulator does not realize communication range and interference range concepts when INET and crSimulator are compared. These properties related to physical layer are controlled by software. Thus, packets are not considered to collide when the PU starts transmitting while the SU is transmitting on the same channel in crSimulator. For this reason, as a future work, we plan to add spectrum sensing module and multi-channel on INET framework. It brings complex programming to develop a new spectrum sensing mechanism and to incorporate the modules from the crSimulator into INET.

Our CRN models provide users to simulate their CRN studies on OMNET++. The developed spectrum sensing mechanisms in [12, 14-15] and spectrum sharing technique in [19] can be implemented and simulated in our models. By developing new algorithms, different channel selection techniques can be adapted and new MAC protocols can be integrated for spectrum access.

Unlike [34-38], PU is no longer used as a jammer, the effects of PU emulation attackers in CRN can be investigated on both PU and SU transmission.

The theoretical studies related to the protection of PUs in [39-42], can be simulated using our simulation CRN models. This allows theoretical studies to be proved practically.

The two-channel CRN model we have developed allows theoretical multi-channel CRN studies such as [43] to simulate.

The CRN models provide an infrastructure for different CRN applications. Complex networks can be created and simulated using the developed models. Thus, the dynamic use of cognitive radio in larger areas can be realized. CRN models can be used as the base structure in routing protocols and multi-hop wireless networks. The study in [44], that simulation is referred to as future work, can be simulated using the models we have developed.

## REFERENCES

1. Engelman R, Abrokwah K, Dillon G. Report of the spectrum efficiency working group. Federal Communications Commission Spectrum Policy Task Force 2002 [cited 2022 11 January]. Available from: [https://transition.fcc.gov/sptf/files/SEWGFfinalReport\\_1.pdf](https://transition.fcc.gov/sptf/files/SEWGFfinalReport_1.pdf).
2. Čabrić D, Mishra SM, Willkomm D, Brodersen RW, Wolisz A. A Cognitive Radio Approach for Usage of Virtual Unlicensed Spectrum. *14th IST Mobile Wireless Communications Summit*. 2005.
3. Joseph Mitola Iii. An Integrated Agent Architecture for Software Defined Radio. PhD, Royal Institute of Technology, 2000.
4. Akyildiz IF, Lee WY, Vuran MC, Mohanty S. Next generation/dynamic spectrum access/cognitive radio wireless networks: A survey. *Computer Networks*. 2006;50(13):2127-2159.
5. Wang B, Liu KJR. Advances in cognitive radio networks: A survey. *IEEE Journal on Selected Topics in Signal Processing*. 2011;5(1):5-23.
6. Biglieri E, Goldsmith AJ, Greenstein LJ, Mandayam NB, Tpoor HV. *Principles of cognitive radio*. New York: Cambridge University Press; 2009.
7. Song M, Xin C, Zhao Y, Cheng X. Dynamic spectrum access: From cognitive radio to network radio. *IEEE Wireless Communications*. 2012;19(1):23-29.
8. MG, Navaie K, Yanikomeroglu H. Access strategies for spectrum sharing in fading environment: Overlay, underlay, and mixed. *IEEE Transactions on Mobile Computing*. 2010;9(12):1780-1793.
9. Zareei M, Muzahidul Islam AKM, Baharun S, Vargas-Rosales C, Azpilicueta L, Mansoor N. Medium access control protocols for cognitive radio ad hoc networks: A survey. *Sensors*. 2017;17(9):2136.
10. Tomar GS, Bagwari A, Kanti J. Introduction to cognitive radio networks and applications. *Introduction to Cognitive Radio Networks and Applications*. 2016: 1–321.

11. Javed JN, Khalil M, Shabbir A. A Survey on Cognitive Radio Spectrum Sensing: Classifications and Performance Comparison. *3rd International Conference on Innovative Computing (ICIC)*; 2019:IEEE.
12. Upadhyay S, Deshmukh S. Blind parameter estimation based matched filter detection for cognitive radio networks. *2015 International Conference on Communication and Signal Processing (ICCSP)*; 2015:IEEE.
13. Chaudhary N, Mahajan R. Spectrum Sensing Techniques in Cognitive Radio Networks: Challenges and Future Direction. *Proceeding of Fifth International Conference on Microelectronics, Computing and Communication Systems*. 2021:451-458.
14. Miah MS, Schukat M, Barrett E. Sensing and throughput analysis of a MU-MIMO based cognitive radio scheme for the Internet of Things. *Computer Communications*. 2020;154: 442-454.
15. Yue WJ, Zheng BY, Meng QM, Yue WJ. Combined energy detection and one-order cyclostationary feature detection techniques in cognitive radio systems. *Journal of China Universities of Posts and Telecommunications*. 2010;17(4):18-25.
16. Mohammad A, Awin F, Abdel-Raheem E. Case study of TV spectrum sensing model based on machine learning techniques. *Ain Shams Engineering Journal*. 2022;13(2).
17. Kaur R, Buttar AS, Anand J. Spectrum Sharing Schemes in Cognitive Radio Network: A Survey. *Proceedings of the 2nd International Conference on Electronics, Communication and Aerospace Technology (ICECA)*; 2018:IEEE.
18. Yücek T, Arslan H. A survey of spectrum sensing algorithms for cognitive radio applications. *IEEE Communications Surveys and Tutorials*. 2009;11(1): 116-130.
19. Wang H, Gao L, Gan X, Wang X, Hossain E. Cooperative spectrum sharing in cognitive radio networks: A game-theoretic approach. *IEEE International Conference on Communications*; 2010:IEEE.
20. Ridhima, Singh Buttar A. Fundamental Operations of Cognitive Radio: A Survey. *Proceedings of 2019 3rd IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT)*; 2019:IEEE.

21. Ahmad A, Ahmad S, Rehmani MH, Hassan NU. A survey on radio resource allocation in cognitive radio sensor networks. *IEEE Communications Surveys and Tutorials*. 2015;17(2):888-917.
22. Usman M, Koo I. Throughput Maximization of the Cognitive Radio Using Hybrid (Overlay-Underlay) Approach with Energy Harvesting. *Proceedings - 12th International Conference on Frontiers of Information Technology*; 2014:IEEE.
23. Jasbi F, So DKC. Hybrid Overlay/Underlay cognitive radio network with MC-CDMA. *IEEE Transactions on Vehicular Technology*. 2016;65(4):2038-2047.
24. Alnwaimi G. Adaptive Underlay/Interweave Transmission Protocol for Cognitive Radio Networks. *Iranian Journal of Science and Technology - Transactions of Electrical Engineering*. 2021;45(4):1-11.
25. Hattab G, Ibnkahla M. Multiband spectrum access: Great promises for future cognitive radio networks. *Proceedings of the IEEE*. 2014;102(3):282-306.
26. Kakkavas G, Tsitseklis K, Karyotis V, Papavassiliou S. A Software Defined Radio Cross-Layer Resource Allocation Approach for Cognitive Radio Networks: From Theory to Practice. *IEEE Transactions on Cognitive Communications and Networking*. 2020;6(2):740-755.
27. Abdullah, M. Wireless lan medium access control (mac) and physical layer (phy) specifications. *IEEE*; 1997:IEEE.
28. Quach MT, Ouattara D, Krief F, Khalife H, Chalouf MA. Overlap regions and grey model-based approach for interference avoidance in cognitive radio networks. *International Conference on Ubiquitous and Future Networks (ICUFN)*; 2013:IEEE.
29. Mohammed, E. Performance Analysis of WRAN at Physical Layer of IEEE 802.22 standardization. *International Journal of Computer Applications*. 2016;133(14):4-8.
30. Liu Y, Hu S, Xiao Y, Liu X, Qu G, Kim K. CSMA/CA-based MAC protocol in Cognitive Radio network. *IET 3rd International Conference on Wireless, Mobile and Multimedia Networks (ICWMNN 2010)*; 2010:IET.

31. Dong Q, Chen Y, Li X, Zeng K. A Survey on Simulation Tools and Testbeds for Cognitive Radio Networks Study. *arXiv*. 2018; *preprint arXiv:1808.09858*.
32. Bani K, Kulkarni V. *Simulation and analysis of IEEE 802.22 cognitive radio network. 2020 International Conference on Convergence to Digital World - Quo Vadis, (ICCDW)*; 2020:IEEE.
33. MiXiM; [cited 2022 8 January]. Available from: <http://mixim.sourceforge.net/>.
34. Caso G, de Nardis L, Holland O. Simulation of Cognitive Radio Networks in OMNeT++. *Cognitive Radio and Networking for Heterogeneous Wireless Networks*. 2015;291-313.
35. Marinho J, Monteiro E. Cognitive Radio Simulation based on OMNeT++/MiXiM. *11<sup>th</sup> Conferência sobre Redes de Computadores*; 2011: CRC.
36. Khan SN, Kalil MA, Mitschele-Thiel A. CrSimulator: A discrete simulation model for cognitive radio adhoc networks in OMNeT ++. *Proceedings of 2013 6th Joint IFIP Wireless and Mobile Networking Conference (WMNC)*; 2013:IEEE.
37. Noor NM, Din NM, Ahmed E, Kadir ANA. OMNET++ based cognitive radio simulation network. *2016 7th IEEE Control and System Graduate Research Colloquium (ICSGRC)*; 2017:IEEE.
38. Loubser CJ, Swart TG. Cognitive radio DAB MAC protocol performance using a CR specific simulator and Software Defined Radio. *IEEE AFRICON: Science, Technology and Innovation for Africa*. 2017; 139-144.
39. Ning B, Yang S, Liu L, Lu Y. Resource allocation for OFDM cognitive radio with enhanced primary transmission protection. *IEEE Communications Letters*. 2014;18(11):2027-2030.
40. Xu W, Yuan W, Shi Q, Wang X, Zhang Y. Distributed Energy-Efficient Cross-Layer Optimization for Multihop MIMO Cognitive Radio Networks with Primary User Rate Protection. *IEEE Transactions on Vehicular Technology*. 2017;66(1):785-797.

41. Shi Z, Teh KC, Li KH. Energy-efficient joint design of sensing and transmission durations for protection of primary user in cognitive radio systems. *IEEE Communications Letters*. 2013;17(3):565-568.
42. Zhang W, Yeo CK, Li Y. A MAC sensing protocol design for data transmission with more protection to primary users. *IEEE Transactions on Mobile Computing*. 2013;12(4):621-632.
43. Sun W, Liu X, Zheng K, Xu Y, Liu J. Spectrum Utilization Improvement for Multi-Channel Cognitive Radio Networks with Energy Harvesting. *International Conference on Networking and Network Applications (NaNA)*. 2021:IEEE.
44. Rai P, Ghose M, Sarma H. Game Theory Based Energy Efficient Routing in Cognitive Radio Wireless Sensor Networks. *International Symposium on Smart Electronic Systems (iSES) (Formerly iNiS)*. 2020:IEEE.
45. CsmacaMac; [cited 2022 16 January]. Available from: <https://doc.omnetpp.org/inet/api-current/neddoc/inet.linklayer.csmaca.CsmacaMac.html>.
46. AckingMac; [cited 2022 16 January]. Available from: <https://doc.omnetpp.org/inet/api-current/neddoc/inet.linklayer.acking.AckingMac.html>.
47. OMNET INET Framework; [cited 2022 9 January]. Available from: <http://inet.omnetpp.org/>.
48. Standard for Information technology Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. *IEEE Std 802.11-2016 (Revision of IEEE Std 802.11-2012)*; 2016:IEEE.
49. Ye F, Sheu ST, Chen T, Chen J. The impact of RTS threshold on IEEE 802.11 MAC protocol. *Journal of Science and Engineering*. 2003;6(1):57-63.
50. Yu C, Lee B, Kalubandi S, Kim M. *Medium access control mechanisms in mobile ad hoc networks*. Mobile Computing Handbook; 2004.