Implementation and Performance Analysis of Probabilistic Cognitive Relaying Communication Demo

Amith Khandakar  
Electrical Engineering Department, Qatar University  
P.O. Box-2713, Doha, Qatar

Amr Mohamed  
Computer Science and Engineering Department, Qatar University  
P.O. Box-2713, Doha, Qatar

Abstract—Cognitive Relaying is the concept where secondary users (SUs) help primary users (PUs) by relaying its traffic. This paper aims to propose the possibility of studying and comparing the performance of cognitive relaying using an experimental framework developed with USRP2. Probabilistic Relaying and Scheduling is practically implemented and its effect on adjusting the delay and throughput of PU and SU are investigated. Step by Step implementation of a cooperative protocol, where SU helps the PU by relaying its data or its own data based on an adjustable scheduling probability. Adjustable admission control is also introduced in the protocol so that PU data that can be relayed will be introduced in the queue of the SU with certain probability. The effect of varying both these probabilities, service and admission is studied in the setup. Finally, the practical results are verified with the simulation and theoretical results, and a conclusion is drawn from the results of the combined effects of probability of relaying and scheduling on the performance of the PU and SU in a cognitive relay environment.

Keywords—Cognitive relaying; GNU radio; probabilistic relaying; USRP2

I. INTRODUCTION

As wireless communication has become the de facto standard for our growing and diverse demands, there’s a need to use the spectrum as efficiently as possible to accommodate future innovations. To do that we need to analyze the spectrum carefully and deduce conclusions that will help us make the spectrum utilization process more efficient. The electromagnetic radio spectrum can be considered as a natural resource and its use by various transmitters and receivers is governed by the different regulatory authorities and agencies. CR (Cognitive Radio) provides a unique solution for the spectrum utilization problem, as it is a smart system which changes itself to minimize, limited resource consumption. The issue of Spectrum scarcity in addition to under-utilization of the licensed spectrum [1] inspired the cognitive radios concept [2], [3] targeting at taking advantage of the spectral holes. These spectral holes are silence periods in which the spectrum is idle. The presence of such holes originates from the bursty nature of the sources, where the users who have legitimate access to the system, called primary users (PUs), do not always have data to transmit. That’s why cognitive radio networks have been gaining increasing worldwide interest. The main idea of cognitive radios resides on introducing cognitive secondary users (SUs) capable of sensing the spectrum and exploiting spectral holes to transmit their packets without affecting the performance of PU or in some cases affecting it to certain allowed limits. In other words, the utilization of the spectrum by the system is improved without disturbing the quality of service (QoS) requirements of the PUs [4].

Cooperative communication in wireless networks has been widely analyzed [5] [6] [17] [18] [19] [20]. Due to the broadcast nature of wireless channels, cooperation is possible and is needed as a single transmission can be heard by different nodes within a locality and data can be lost between transmitter and receiver but can be successfully received by nodes in between who can in turn relay the lost traffic to the receiver. In [5], the authors sketch some tactics employed by the cooperating radios, including amplify-and-forward and decode-and-forward schemes. Performance characterizations in terms of outage events and probabilities were also developed by them. In [7], many partner cooperative transmission protocols are proposed and evaluated using Zheng-Tse diversity-multiplexing trade-off [8]. Ibrahim et al provided a symbol error rate analysis for decode and forward cooperation protocol in [9], which is as a starting point for a relay selection mechanism developed and analyzed in [10]. Cooperative communication can be like the notion of spatial diversity where multiple antennas are used to achieve spatial diversity in single communication links [11], [12], and in the case of cooperative communication multiple nodes are used.

Cooperation in cognitive radio networks not allows SUs to make use of the idle time slots for transmitting their own data but also to relay the lost packets of PU, confirming that cooperation in cognitive radio networks is beneficial in all possible ways [24]. Retransmissions by the PU are reduced as the SUs support by delivering their packets to the destination, which provides more timeslots for SU to deliver its own packets, and this process is termed as cognitive relaying. However, this is subjected to SU having a better channel to destination. Cognitive relaying helps in making more time slots available for SU’s to transmit it data after fulfilling the demands of the PU’s.

Many previously works in cognitive and cooperative relaying deal from a physical layer perspective and not much physical implementation and analysis of actual results was
done. Nevertheless, the work in the paper will be focusing on:

(i) Preparing an actual test setup using Software Defined Radio (SDR-USRP) with GNU Radio interface [21] (ii) engaging cooperation (Cognitive Relaying) at MAC layer and exploring its potentials in terms of delay and throughput, specifically the consequence of introduced Scheduling probability on the routine of PU.

Many works are done in the field of performance comparison in Cooperation Cognitive relaying like in [13], power allocation at the SU, which has the capability of relaying the packets of the PU, is done with the objective of maximizing the stable throughput of the cognitive link for a fixed throughput selected by the primary link. In [14], a fraction of the PU bandwidth is allocated to SU transmission based on gain achieved from cooperation. Many protocols are studied in [15] which allow collaboration between a PU and several SUs. [15] Which inspires the opportunity for simultaneous transmission of primary and secondary data by means of dirty-paper coding [16]. Performance gains in terms of stable throughput region and average delay are demonstrated in [22] and is used in this paper.

In this paper, a detailed analysis of Cooperative Relaying for cognitive radio is done by developing a MAC protocol from scratch and the results compared with the theoretical results of the MAC implementation in [22]. The MAC protocol relies on the SU having two queues, one for its own packets and the other for the PU’s relayed packets. Investigational setup using USRP and different SDR platforms (such as Matlab, Gnu Radio, Lab View) have been done for spectrum sensing and cooperative relaying [23] [25] but did not show the step by step MAC protocol implementation and introduce the 2 conditional probabilities as the one discussed in the implementation stated in this paper.

Contrasting to the predictable relaying that allocates full priority to the relay queue, an investigation is done with a customizable randomized cooperative policy with probabilistic relaying and the theoretical background for such an implementation is obtained from the work of [22]. According to the proposed policy in [22], admission control is introduced at the relay queue of the SU, where a PU’s packet that is dropped on the way to the destination, can enter the relay queue with probability $P_s$ (Probability of admission). Besides that, when the SU decides to transmit in a detected idle time slot, it serves the relay queue with a probability $P_r$ or the queue of its own data (with a probability $(1-P_s)$). Thus, the delays of PU and SU with the help of the introduced probabilities can be used to enhance the performance of either of them (it can be controlled to prioritize PU performance) or both. Thus, the scheme could be adjusted according to the weights of the planned applications running at both the PU and SU. The PU and SU throughput and delay trade-offs are studied. The PU and SU delay is then compared with the simulation results. The significance of the proposed policy lies in the varying ability, whereby a variety of objectives could be realized via performing constrained optimizations over the degrees of freedom of the system represented by the relay queue admission probability, $P_a$, and the queue selection probability, $P_s$. The main contributions of this work are summarized as follows:

- Established an investigational background for reviewing cognitive relay scenario, making use of scheduling and admission policies.
- Established a MAC layer protocol for reliable and synchronized communication between PU Transmitter, Receiver and the Relaying SU.
- Analyzed how the admission and scheduling probabilities could be tuned to improve SU performance while not affecting the PU's performance.
- Steered a comparative study between implementation and theoretical results [22] of the cognitive relaying scenario to verify the feasibility of the investigational background.

The rest of this paper is organized as follows. Section II presents the system model along with the test setup description. Section III describes the results obtained from the test setup and compares it with theoretical results in [22] and finally the conclusion is drawn in section IV.

II. SYSTEM MODEL AND DESCRIPTION OF THE TEST SETUP

A. Description of the Test Setup

Fig.1 summarizes the test setup using 4 USRP’s:

- Syncing Node: This node synchronizes the time slots of the SU, PU TX and PU RX, by transmitting Sync packets periodically for syncing the time slots, as Time Division Multiple Access MAC protocol has been used.
- PU TX is the Primary User which is transmitting.
- PU RX is the Primary User which is receiving.
- SU is the Secondary User which behaves as the cognitive relay based on the introduced probabilities.

In the paper and work in [22] the probability that a packet is decoded without error determines the channel quality as the channel gain and noise processes are assumed not to be random, and the nodes are fixed. A successful transmission requires receiving the entire packet without error; otherwise, the packet is discarded. Resulting in the channel reception probability being a constant value between 0 and 1. As Time Division Multiple Access is used, the SU transmits if and only if it finds any time slot idle ensuring that the system cannot have collisions for multiple nodes transmitting at the same instance. Hence, the only reason for packet loss is the channel impairments since no collisions are allowed. These channel impairments are typically caused by fading, shadowing, signal attenuation and additive noise. The event that causes packet loss is the channel outage event, which is characterized as the signal-to-noise ratio (SNR) at the receiving node being below a pre-specified threshold. This threshold is the minimum value of the SNR required by the receiver to perform an error-free decoding. Let $P_{PS}$, $P_{PD}$ and $P_{SD}$ represent the probability of successful transmission between the PU TX and SU, PU TX and PU RX, and the SU and PU RX, respectively. Throughout the paper, it is assumed that $P_{PD} < P_{SD}$, as we want the SU to act as a cooperative relay which can happen only if it has a better channel to the destination compared to the PU.

www.ijacsa.thesai.org
Acknowledgements (ACKs) sent by the destination and the SU for overheared primary packets, are assumed instantaneous and can be heard by all nodes error-free. The numerical values of the probability of successes are stated below:

- $P_{ps}$ is assumed to be 0.95 (very good channel condition)
- $P_{sd}$ is assumed to be 0.95 (very good channel condition)
- $P_{pd}$ is assumed to be 0.7 (Bad in comparison to the other channel condition)

$P_{ps}, P_{sd} \gg P_{pd}$, which implies that the channel condition between PU TX and PU RX via SU is better than the channel condition between PU TX and PU RX directly.

$P_{SUD}$ denotes the probability of successful transmission between SU and PU RX (it will also behave as the receiver for the SU) and it is expected to be 0.9 (very good channel)

$P_{a}$ denotes the Probability of Admission, which determines the probability if SU will allow the PU TX relay data in the PU Queue of SU. In other words, it can be the degree of Cognitive Relaying permissible.

$P_{s}$, denotes the Probability of Scheduling, which determines the probability that SU during the empty time slots (when it can use the channel) will relay the dropped PU TX data and not its own data.

$\lambda_{p}$ denotes the rate at which packets arrive at the PU TX queue according to Poisson distribution. It is set to 0.4

$\lambda_{s}$ denotes the rate at which packets arrive at the SU queue according to Poisson distribution. It is set to 0.4

B. MAC Description

The MAC is developed using similar strategies and theory from [17] [18] [19] [20] [23] and is described below followed by step by step development description:

- If the transmitted packet from PU is successfully interpreted by the receiver PU then it broadcasts an ACK. This packet exits the system. If it is not received by the receiver PU but was successfully received by the SU then the SU buffers the packet with the admission probability ($P_{a}$) or discards it ($1 - P_{a}$).

  - Once a packet is buffered by the SU, it sends back an ACK to declare successful reception of the PU’s Packet. It becomes the obligation of the SU to deliver the packet to the destination.
  - In the worst case when the packet is neither successfully received by the Destination nor by the SU, it is retransmitted by the PU Sender in the next time slot.
  - When a time slot does not have any PU activity, the SU uses it to transmit either a packet from Primary User Queue (with probability $P_{s}$), refer Fig 1, or from Secondary User Queue (with probability ($1 - P_{s}$)).
  - Whenever a packet is not exited from the system upon reception of ACK then it is kept at queue (either sender PU or SU) for later retransmission.
  - In case there is no packet to be sent by either the sender PU or SU then the channel remains idle.

The proposed policy of the MAC is non-work-conserving for the simplicity of implementation. A system is considered work-conserving if it does not remain idle whenever it has packets [19]. If the working scheme is understood then it can be noticed that in the test setup, whenever there is an idle slot the SU may have something to transmit either the PU TX data that it would relay or its own data and if neither queue have any packet, the slot remains idle. It is done for reducing the complexity of the implementation and the performance of such an implementation being work conserving is a future work and can provide interesting results.

C. States and their Description

In Fig. 2 and Fig. 5, the PU TX has four States and it can be in any three of them, based on conditions and MAC protocol design, stated briefly in the previous section. Each state is allocated a period of 1 second in a frame. Each frame denotes a packet being transmitted either by PU TX or SU. The time for states and frames are kept constant in the experiment and can be varied (made smaller) in practical applications:
1) **Transmit state:** The Primary User Transmitter is transmitting its data in this state. The PU is transmitting whenever it has a packet in its queue determined by $\lambda_p$. It transmits a single packet during this state before moving the state of PuAck.

2) **PuAck state:** PU TX delays for the Acknowledgment (ACK) from the PU RX before going to the Free State. In case of no (ACK) it goes to the SuAck state where it waits for the acknowledgement from SU that it would relay the packet to the PU RX whenever the channel is idle from PU activity. PU TX goes to either Free or SuAck State at the end of this state.

3) **SuAck state:** After the timeout of PuAck state and not receiving any acknowledgement from PU RX, PU TX waits for the confirmation from the SU (in the form of acknowledgement) that it would relay the packet to the PU RX whenever the channel is free from any PU activity. The acknowledgement depends on $P_a$. This state marks the end of the frame and the packet is transmitted in the next frame by the SU depending on the activity of the channel.

4) **Free state:** Having received the acknowledgement from PU RX, there is no activity in this state and thus it will be the Free State. This state marks the end of the frame and more packets, if available in PU queue, is transmitted in the next frame.

5) **Sync state:** This state is to synchronize the Time Slots of all the USRP’s-PU TX, PU RX and SU. A dedicated USRP is always broadcasting the Sync packet to sync any new USRP to the time slots of the remaining communicating USRP’s. Once synced then the USRP’s continue their states as per the protocol.

As shown in Fig.3 and Fig. 5, the SU has six States and it can be in any four of them, based on conditions and MAC protocol design, stated briefly in the previous section. It follows the similar duration for states and frames as explained earlier.

6) **Sense state:** Based on a sensing algorithm developed in the MAC layer, where the SU senses the spectrum is occupied if it receives any packet with the Packet Type Field of 2 (the Packet Format and Packet Type Field is described later). The SU seeks for this information in the initial 300 ms of the state. If the information is found (PU activity sensed) then it transits to the Receive state or else to the TXPU/TXSU state (the selection between TXPU/TXSU is dependent on the packets in both the queues of SU and $P_s$, explained in detail later).

7) **Receive state:** Upon sensing activity, SU moves to this state and stores the PU packet in one of the two queues it has, i.e. PU Relay Queue, depending on $P_s$. To avoid blocking the channel it does not acknowledge the reception.

8) **TX PU/TX SU state:** Upon not sensing activity, SU enters this state where it can either transmit its own data (if it has and with probability $(1- P_s)$) or the queued PU Data (which were not acknowledged by PU RX and was allowed based on $P_a$ with probability of $P_a$.

As shown in Fig.5, the Sense State and Receive State or TX PU/TX SU State collectively have 1 sec duration. After these states, SU enters the RX PuAck State.

9) **RX PuAck state:** SU waits for any acknowledgment coming from the PU RX or from the SU RX. Upon reception of acknowledgement the respective packet is enqueued from the respective queue (either PU Relay Queue or its own Queue). Obviously, it won’t receive both the acknowledgement at the same time as SU is not allowed to use the channel if there is PU activity. Depending on the acknowledgement from PU RX, it transits to either the Free State or the TX SuAck State.

10) **TX SuAck state:** Upon not reception of acknowledgement of reception for PU RX, SU having received the packet during Receive State (depending on $P_s$) acknowledges PU TX that it would relay the packet for it whenever the channel is idle from PU's activity This state marks the end of frame.

11) **Free state:** Having received the acknowledgement from PU RX, there is no activity in this state and thus it will be the Free State. This state marks the end of the frame.

12) **Sync state:** This state is like the Sync state explained earlier.
which is of duration 3 seconds for each state.

As stated in Lemma 7 in [22], in the cooperative relaying of PU data by the USRP, the PU TX has 2 states with duration of 1 second and should repeat one of the states to have at least 3 states to fill up the frame (which is of duration 3 seconds). The states are based on conditions and MAC protocol design, stated briefly in the previous section.

13) Receive state: The USRP behaving as PU RX is also a receiver for the SU and is done for the experiment purpose, as the same USRP is used as receiver for both. In this state, PU RX waits for the reception of either the PU TX packet or SU packet (obviously both PU TX and SU are not sending together and Packet Type field in the packet format which help the USRP to identify if it PU TX or SU packet and act accordingly and separately without interfering with the respective tasks). Unlike other states in PU TX and SU, the PU RX remains in this state till it receives any packet from either PU TX or SU, else it moves to the Transmit State.

14) Transmit state: SU sends acknowledgement in this state to either PU TX or SU for their packets received.

15) Sync state: This state is like the Sync state explained earlier.

As shown in Fig. 5, the description and duration of the time slots for each state of the PU TX, PU RX and SU are stated. In Fig. 6, the Packet format used in the test setup is stated.

### III. COMPARISON OF RESULTS

The theoretical background of the setup is achieved from [22]. As shown in Fig. 7, the experimental results and the simulation results (using the same MAC protocol design) show similar trends where the Average PU packets Delay is constant for \( (1-P_a) = 0.3 \) and increases monotonically with \( P_a \) when \( (1-P_a) = 0.5 \) and decreases monotonically with \( P_a \) when \( (1-P_a) = 0 \). As stated in Lemma 7 in [22], the cooperation is beneficiary to the PU if \( 1-P_s \geq (1-P_a)/P(SD) \). This could be further explained with the reasoning that the quality of the channel of the direct link between the PU and the destination is inversely proportional to the benefits of the cooperation since the probability of packet successful delivery in the direct link increases due to the better channel. It is also found that the threshold of \( (1-P_a) \) values below which cooperation is beneficiary from the PU’s point of interest, is given by \( (1-P_a/P(SD)) \) [22], and it is 0.3 in the experimental setup which further validates the result in Fig. 7. It is also noticeable that the degradation in the PU’s delay with the increase of \( (1-P_s) \) is negligible at lower values of \( P_a \), which also shows the reputation of parameter \( P_a \) in adjusting the systems performance.

Similarly, Fig. 8 shows that the Average SU packets Delay is decreasing with increasing \( (1-P_a) \), which confirms that SU is always benefits from increasing \( P_a \), as it prevents a lot of retransmission from the PU by relaying the dropped packets for it, which in turn provides it with more idle time slots. These idle time slots can be used by SU for transmitting its own data as well depending on \( P_a \). At fixed \( P_a \), SU’s packets have lower delay at higher values of \( (1-P_a) \) as more scheduling of its own data. As seen in Fig. 9, the average throughput per cycle for PU packets increases with \( P_a \) when \( (1-P_a) < (1-P_a)/P(SD) \) i.e. \( (1-P_a) < 0.3 \). This is due to the cooperative relaying of PU data by the SU.
IV. CONCLUSION AND FUTURE WORK

The paper has demonstrated a practical implementation of a Cognitive Relay environment using USRP2 and GNU Radio. Certain Probability parameters are introduced to meet the requirements of having a trade-off of Throughput and Delay for both the PU and SU. The findings were verified [22] using the Cognitive Relay framework where increasing \((1-P_s)\) is always in favour of the SU as opposed to the PU in terms of both throughput and delay. This paper is part of the Master Thesis submitted [24]

ACKNOWLEDGEMENT

This work was supported by NPRP from the Qatar National Research Fund (a member of Qatar Foundation) under Grant 7-684-1-127. The statements made herein are solely the responsibility of the authors.

REFERENCES


www.ijacsa.thesai.org