A Spectrum Sharing Framework for Intelligent Next Generation Wireless Networks

Felipe A. P. de Figueiredo*, Xianjun Jiao*, Wei Liu*, Ruben Mennes†, Irfan Jabandžić*, and Ingrid Moerman*

*Ghent University - imec, IDLab, Department of Information Technology, Ghent, Belgium
†Department of Mathematics and Computer Science, University of Antwerp - iMinds, Antwerp, Belgium
Email: [felipe.pereira, xianjun.jiao, wei.liu, irfan.jabandzic, ingrid.moerman]@ugent.be, †ruben.mennes@uantwerpen.be

Abstract—**The explosive emergence of wireless technologies and standards, covering licensed and unlicensed spectrum bands has triggered the appearance of a huge amount of wireless technologies, with many of them competing for the same spectrum band instead of harmoniously sharing it. Unfortunately, the wireless spectrum is a scarce resource, and the available frequency bands will not scale with the foreseen demand for new capacity. Certain parts of the spectrum, in particular the license-free ISM bands, are overcrowded, while other parts, mostly licensed bands, may be significantly underutilized. As such, there is a need to introduce more advanced techniques to access and share the wireless medium, either to improve the coordination within a given band, or to explore the possibilities of intelligently using unused spectrum in underutilized (e.g., licensed) bands. Therefore, in this paper, we present an open source SDR-based framework that can be employed to devise disruptive techniques to optimize the sub-optimal use of radio spectrum that exists today. We describe three use cases where the framework can be employed along with intelligent algorithms to achieve improved spectrum utilization. Additionally, we provide several experimental results showing the performance and effectiveness of the proposed framework.**

Index Terms—**Next Generation Wireless Networks; Cognitive Radios; Collaborative Intelligent Radio Networks; Spectrum Sharing; Coexistence; Experimental Evaluation.**

I. INTRODUCTION

The demand for wireless broadband services is insatiable. Current wireless networks can not offer spectrum bandwidth and network capacity to meet the growing traffic demands (e.g., consumers, enterprises, vertical industries, service providers, etc.) forecast for the coming years. According to [1], wireless networks will see an increase in usage by 47 % CAGR by 2021, reaching staggering 49 Exabytes per month, when for example, speeds are expected to reach peaks of 10 Gbps. Additionally, devices ranging from smartphones to wearable fitness recorders to smart kitchen appliances are voraciously competing for bandwidth. With everything considered, around 50 billion wireless devices are expected to be competing for access to wireless communications networks in the next coming years. By 2030, the demand for wireless access is foreseen to be 250 times what it is today [1].

Next generation wireless networks are expected to provide broadband access wherever needed and also support a diversified range of services including everything from self-driving cars to virtual reality, robotic surgery and Internet of Things (IoT) [2, 3]. Connections in the order of one to dozen of Gbps to vehicles, high speed trains, data-intensive services (e.g., augmented reality, immersive 360° experiences, etc.) are some of the applications that will drive the demand for larger coverage area, lower latency and higher capacity at reduced cost in next generation wireless networks [4]. Unfortunately, the wireless radio spectrum is a scarce resource, and the available wireless bandwidth does not scale with the needed wireless bandwidth [5]. Hence, as can be seen, the development of technologies that can tackle the ever increasing traffic and service demands while combating the imminent scarcity of spectrum bandwidth is of vital importance for next generation wireless networks.

The development of such technologies will be driven by three broad use cases families, namely, Enhanced Mobile Broadband (eMBB), Ultra Reliable Low Latency Communications (URLLC), and Massive Machine Type Communications (mMTC). They aim at improving performance, scalability and (cost/energy) efficiency of the current wireless networks such as LTE. Therefore, these use cases and their direct requirements will demand huge improvements in comparison with the previous generation of IMT systems [6]. Next, we briefly discuss each one of them and list some possible applications [6, 7].

- **eMBB**: this use case focus on enhancements to the data rate, user density, latency, capacity and coverage of the actual wireless broadband access networks [8]. Some of its applications are: (a) High-speed mobile broadband; (b) Augmented Reality; (c) Virtual Reality; (d) Smart office environment, i.e. all devices wirelessly connected; (e) Pervasive video, e.g., high resolution video communications, ultra high definition multimedia streaming, etc.
- **URLLC**: this use case enables devices and machines to establish communication links with ultra-reliability, high availability and very low latency, which makes it ideal for the following applications [9]: (a) Wireless Industrial Control; (b) Factory Automation; (c) Remote Surgery; (d) Cellular Vehicle-to-Anything (C-V2X) communications; (e) Drone communications; (f) Smart Grids; (g) Public Safety.
- **mMTC**: this use case focus on enabling communications between devices that are low-cost, massive in number and battery-driven [7, 8]. It is intended to support applications like: (a) Smart metering; (b) Smart Cities; (c) Asset tracking; (d) Remote monitoring, e.g., field and body

sensors.

In consequence of these new use cases, the next generation of wireless networks will be required to deliver ultra-fast speeds, very low latency, huge coverage area and excellent reliability to dozen of billions of wireless devices. It is worth mentioning that not all of these requirements have to be met at the same time, as different services have different subsets of QoS requirements. Additionally, different services with different QoS constraints further have to share the same spectrum band.

Some approaches that can be used to cope with the expected traffic increase are: (i) increased/improved spectrum availability (such as introduction of new spectrum bands or more efficient/intelligent use of the available ones) [10]; (ii) introduction of technical enhancements (such as new radio interfaces, codecs, use of multicast transmissions (i.e., simultaneous transmission of data to a group of users through point-to-multipoint communication), reduction of energy consumption, etc.); (iii) new network structures and topologies aiming at increased network density (i.e., the addition of more radio sites to the same geographical area and the consequent shrinkage of cell sizes) [11]; (iv) traffic offloading to less occupied spectrum bands (offloading traffic onto unlicensed bands makes capacity available for other users in the license band and improves user experience for devices being served in the unlicensed band. It could also happen within unlicensed bands, e.g., offloading traffic from 2.4 GHz to 5 GHz band) [12]. As we discuss in the next section, our research focus on making more efficient and intelligent use/sharing of the available spectrum bands.

A. Motivation

Most of today's channel allocations separate wireless systems by splitting the spectrum into fixed and exclusively licensed bands that are assigned over large and geographically defined regions. This approach restricts access to the spectrum in exchange for guaranteed interference-free communications. These allocations of spectrum are human-driven and not adaptive to the dynamics of traffic demand and supply. At any given time, many allocated spectrum bands are unused by their licensees while other bands are completely flooded. For example, a report from the Federal Communications Commission's (FCC) Spectrum Policy Task Force (SPTF) shows that 85 % of current allocated radio frequency bands are either partially or completely unused at different times across geographical areas [13]. This kind of channel allocation scheme tremendously wastes the spectrum capacity and creates unnecessary scarcity [5].

Spectrum sharing, where more than one user shares the spectrum band, either in time and/or space, is one possible and highly viable approach to achieve better spectrum utilization (i.e., combat spectrum bandwidth scarcity) and meet the foreseen increase in traffic demand. Additionally, spectrum sharing can be categorized into two different types: (i) sharing in unlicensed bands; and (ii) sharing in licensed bands. Type (i) can be further split into spectrum sharing in unlicensed

bands (i-A) with and (i-B) without an anchor in licensed bands. Some technologies using sharing type (i-A) are LTE-Licensed Assisted Access (LTE-LAA) and LTE-Unlicensed (LTE-U). LTE-LAA and LTE-U occupies licensed and unlicensed bands at the same time through the use of carrier aggregation. Signaling and traffic with specific Quality-of-Services (QoS) requirements will use licensed bands, while less critical traffic is offloaded onto unlicensed bands [12, 14, 15]. Some technologies employing sharing type (i-B) are Wi-Fi and MulteFire, which is a LTE-LAA based technology that works solely in unlicensed bands without the need for an anchor in licensed band [16]. For Type (i) sharing, simple sense-and-avoid techniques such as Listen Before Talking (LBT) and Carrier Sense Adaptive Transmission (CSAT) are applied in order to comply with regulations [10]. The need for LBT is mandatory in some regions (e.g., Europe and Japan), and not mandatory in other regions (e.g., US and China) [12]. For Type (ii) sharing, the techniques employed are Spectrum Access Systems (SAS) in the US and Licensed Shared Access (LSA) in Europe. These techniques will be deployed to allow unused spectrum bands of incumbents to be shared with licensees. Such networks will not use LBT schemes, but will rely on a central licensing authority to ensure that interference is avoided by setting exclusion/protection zones. Access to shared, licensed-shared and unlicensed spectrum bands will be of huge importance to next generation wireless networks as they have the potential to provide significant gains, both in terms of spectrum efficiency and spectrum reuse. These gains translate into higher data speeds for the users, enables new capabilities such as prioritized guaranteed resources (i.e., guaranteed time/frequency resources that provide each network with QoS similar to that it would get from licensed spectrum use) and new deployment scenarios [17].

Technologies devised for spectrum sharing in unlicensed bands, such as LTE-U, LTE-LAA and MulteFire, avoid interfering with neighbor networks by employing simple sense-and-avoid techniques [14], however, in the foreseen dense and diverse future use of spectrum, these simple schemes will not survive, because only local information about the medium availability (which brings about an issue known in the literature as the *hidden node problem* [14]) and naive avoidance mechanisms are insufficient to maintain end-to-end QoS of a myriad of competing wireless links. On the other hand, technologies for spectrum sharing in licensed bands, such as SAS and LSA, avoid interference by having a central arbitrator to decide how the licensees will access the incumbents' band. However, in order to ensure reliable QoS to incumbents and licensees, these technologies can not depend exclusively on geo-location and spectrum data bases to decide how they grant access the spectrum [18]. These licensed-sharing technologies will require algorithms that provide fair and demand-driven allocation of power and spectrum resources to licensees so that both incumbents and licensees have their QoS requirements met.

Consequently, in order to reap the full capacity out of all types of available spectrum bands (i.e., licensed, unli-

censed or licensed-shared bands) and attain stable and adequate communications links, it is clear from the discussion above that technologies for next generation wireless networks have to go beyond simple interference avoidance, frequency isolation (i.e., fixed channel allocation schemes) and geolocation/data base based access approaches. Radios being devised for next generation networks will have to employ greater intelligence and collaboration to avoid interference while maximizing spectrum usage and capacity. These radios will need to intelligently collaborate with their peers in order to manage and optimize the use of the available spectrum without prior knowledge of each other's operating characteristics [19]. Therefore, collaborative intelligent radios should be capable of collaborating with other previously unknown radios and adapt their communication schemes to optimize the combined wireless spectrum usage across all radios. These radios have to be devised not only to communicate reliably in congested and contested environments but also to share the available spectrum band without any central coordination or spectrum pre-allocation planning across a wide range of heterogeneous radios.

In summary, next generation wireless networks will be required to add advanced intelligent algorithms to their radios so that they can collectively develop strategies that optimize the use of the wireless spectrum in ways not possible with today's intrinsically simple and often centralized approaches. As a consequence, researchers need to rethink the current strategies adopted for spectrum access and come up with new wireless paradigms where radio networks can autonomously collaborate, understand the current state of the available spectrum, and jointly reason how to share it more efficiently, i.e., jointly exploit opportunities to attain optimum/efficient use of the available spectrum bandwidth. Figure 1 depicts one possible outcome of the intelligent collaboration among radios with different needs and characteristics. By applying an intelligent collaborative spectrum sharing scheme, the radios can move from a coarse and often overlapping spectrum distribution (i.e., poor spectrum usage due to the absence of cooperation among radios) to a more fine-grained and aligned spectrum allocation (i.e., optimized spectrum usage).

Intelligent and collaborative spectrum sharing models will drive the research and development efforts towards novel concepts aiming at increasing the spectrum usage efficiency and enhancing the coexistence among heterogeneous wireless networks [20]. These concepts will cover a wide range of topics, including the application of machine learning (ML) and artificial intelligence (AI) algorithms to the sharing problem, new deployment models, dynamic spectrum allocation and regulatory aspects, novel radio access technologies, standalone operation of networks (i.e., operation without mobile infrastructure support), new network architectures, ecosystems and business models, etc.

In recent years, prototyping and experimental validation of innovative wireless technologies have gained importance due to the ever increasing complexity of the wireless ecosystem in emerging next generation wireless scenarios. Such complex

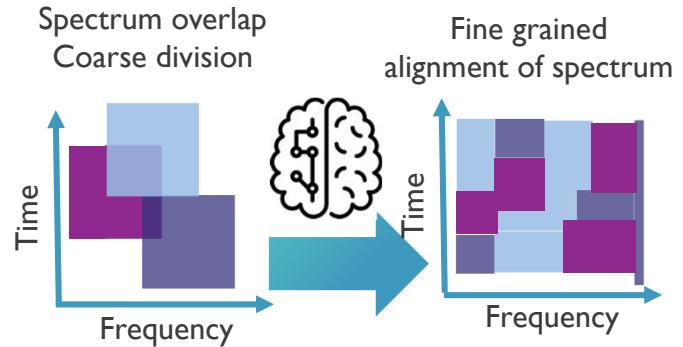



Fig. 1. Collaborative intelligent spectrum sharing.

systems cannot be analyzed anymore with theoretical models or in simulators without applying oversimplified mathematical models and assumptions, far away from limitations imposed by real hardware and real-life environments. Testbeds play a major role in developing and testing new wireless communications technologies and systems and, as with any disruptive technology, prototyping using realistic testbeds is the best way to truly understand the performance trade-offs and limitations. Based on that, several research initiatives are proposing the design of flexible, re-configurable and re-programmable prototyping frameworks and platforms for evaluating, comparing and validating the performance trade-offs of innovative wireless devices, communication techniques, network models, services, etc. in realistic testbed environments.

Software Defined Radios (SDRs) [21, 22] are radio communication systems where several transceiver components, which are typically implemented in hardware (e.g., filters, modulators, demodulators, etc.), are instead, implemented in software on a personal computer or embedded system. SDR platforms provide flexibility in reconfiguration of baseband algorithms, software and reprogramming of Radio Frequency (RF) parameters. Moreover, the concept of SDR is very encouraging for the development of novel wireless communications technologies, as software programming allows much faster development cycles and real-life experiments to be conducted, for example, at local or remote testbeds.

In order to ensure that the next generation of wireless networks can get the most out of collaborative and intelligent spectrum sharing techniques in real-world environments, we present in this paper an open source and runtime configurable SDR-based framework, which is suitable for researching and prototyping novel spectrum sharing and coexistence mechanisms in realistic environments [23, 24]. The framework is composed of three main modules, namely, slot-based physical (PHY) layer, spectrum sensing and collision avoidance (i.e., listen before talk) modules. The proposed framework is implemented using USRP Hardware Driver (UHD) software Application Programming Interface (API) [25] and runs on commercially available off-the-shelf (COTS) hardware devices such as the Universal Software Radio Peripheral (USRP) [26], which is a well-known platform for SDR development.

The main contributions of this paper are summarized next.

- Proposal of an open-source SDR-based framework for intelligent spectrum sharing research in the context of next generation wireless networks. The proposed framework can be used to prototype and experimentally test intelligent algorithms devised to dynamically optimize the use of radio spectrum. We provide detailed explanation on each one of the modules making up the framework. The framework source code can be downloaded from its GitHub project page [23].

- Description of three use cases where the proposed framework can be used to dynamically optimize the use of radio spectrum, combating spectrum scarcity. The use cases show how the framework can be used to leverage intelligent spectrum sharing research for next generation wireless networks.

- Extensive experimental results showing the efficiency of the proposed framework. The results include experiments with filtered-OFDM, closer spectrum coexistence, collision avoidance and measurements of the throughput and CPU/Memory utilization.

This paper is organized as follows. In section II, related works are compared with and distinguished from our work. Section III describes the proposed spectrum sharing framework for intelligent next generation wireless networks. Section IV presents in detail a highly flexible slot-based PHY that can be used in collaborative intelligent spectrum sharing research. Section V describes the RF monitor module and the spectrum sensing algorithm used by this module to assess the medium state. Section VI introduces a totally configurable software based Listen Before Talking (LBT) module. Section VII presents some use cases with suggestions on how the proposed framework could be employed in intelligent next generation wireless networks. In section VIII, we present and discuss the results of several experiments performed with the proposed framework. Finally, section IX presents our conclusions and indicates directions for future work.

II. RELATED WORK

Three of the most well-known open source LTE frameworks are Eurecom's OpenAirInterface (OAI) [27], openLTE [28] and Software Radio Systems' srsLTE [29]. OAI is compliant with LTE release 8.6 and implements only a subset of release 10. Additionally, it only supports 5, 10, and 20 MHz bandwidth and the code structure is complex and difficult to customize. OpenLTE's source code is well organized and can be customized to some extent, however, it lacks detailed documentation, e.g., there is no information on compliance with any 3GPP release and it has a very silent mailing list. Furthermore, it is still incomplete and with several features unstable or under development. On the other hand, srsLTE's source code is well organized with a modular structure, has good documentation, a very active mailing list and can be easily customized. However, it is only compliant with LTE Release 8 and implements a few features of Release 9. These open source frameworks can only be configured through

configuration files (i.e., static configuration) and none of them offers any mechanism or feature for spectrum sharing.

The Amarisoft LTE-100 platform is a commercial and closed source SDR-based LTE network suite. This solution is compliant with 3GPP LTE Release 14 [30], however, as far as we are concerned, it does not support any of the features necessary for the operation in unlicensed bands, which are the base for spectrum sharing approaches. This platform can not be configured in real-time, only allowing file based configuration. The cost of the Amarisoft LTE-100 software suite ranges from 4500 to 8000 € depending on licensing type (fixed or floating) and number of channels. Moreover, as it is a closed-source solution, it can not be modified.

On the other hand there is the National Instruments' (NI) LabVIEW Communications LTE-Advanced Application Framework [31]. This proprietary framework implements a subset of the 3GPP LTE Release 10 and provides support for the FlexRIO PXIe-7975/7976R and USRP RIO devices. The LTE framework is easy to be modified, mainly due to LabView's graphical programming language, allows real-time prototyping and is extensively validated but has no built in coexistence feature. In [32], NI customized the LTE framework in order to implement some LTE-U and LTE-LAA features like discontinuous transmission and LBT. However, the customized framework can not be bought separately from the whole LabVIEW 802.11 and LTE-Advanced Application Frameworks, once it is sold as a Real-time LTE/Wi-Fi Coexistence Testbed. Additionally, the customized LTE framework can only be configured through a graphical user interface (GUI) and only allows threshold and Contention Window (CW) size parameters to be configured in real-time through the GUI. Moreover, this is a quite expensive solution, costing more than $66000.

A framework, named CONTACT, to explore emerging coexistence techniques among multiple Radio Access Technologies (RAT) in both wireless communication and computer networking is presented in [33]. The proposed framework is divided into three layers: radio access, network and control layers. However, the framework only implements a simple preemptive sharing scheme at the radio access layer, i.e., it implements Carrier-Sensing Adaptive Transmission (CSAT) to sense channel usage and adjust the on and off LTE cycling based on Wi-Fi usage [34]. Such schemes do not take collaborative and intelligent spectrum sharing into account and can not be deployed world-wide as some countries require the use of contention based access mechanisms [34]. The framework only allows real-time configuration of the ON/OFF duty cycle, however, the authors do not mention by which means (e.g., through pre-defined control messages over network) it is carried out. Additionally, the proposed framework is not publicly available either as open or closed source.

To the best of the authors' knowledge there is no open source framework that offers researchers and practitioners the flexibility and the necessary modules (i.e., LBT and RF monitor) to devise, implement and experimentally test novel spectrum sharing schemes. Additionally, none of the

TABLE I
COMPARISON OF THE RELATED FRAMEWORK SOLUTIONS.

Framework	Open Source	Real-time configurability through control messages	RF Monitor	LBT	Discontinuous Tx	Cost
OAI [27]	Yes	No	No	No	No	-
openLTE [28]	Yes	No	No	No	No	-
srsLTE [29]	Yes	No	No	No	No	-
CONTACT [33]	No	Not informed	No	No	Yes	-
Amarisoft [30]	No	No	No	No	No	4500 to 8000 €
NI's LTE Framework [31]	No	No	No	No	No	> $66000

above mentioned solutions allows users to easily plug-in other modules, receive information and change parameters in real-time through pre-defined messages over the network. Table I summarizes the comparison of characteristics presented by each one of the related solutions.

Next, we list some related simulation works that can make use of the proposed framework to also validate the ideas through experimental validation.

In [35] the authors propose the use of particle swarm optimization to learn the behavior of spectrum usage, helping to allocate spectrum dynamically. They also introduce a intelligent and optimal relay selection algorithm for effective selection of relay nodes, which enhances amplify and forward relay selection algorithm using intelligent agents.

A spectrum allocation solution using multi-agent system co-operation that enables secondary users to utilize the amount of available spectrum, dynamically and cooperatively is proposed in [36]. The agents are deployed on primary and secondary users that cooperate to achieve a better spectrum usage.

A Q-Learning based dynamic duty cycle selection mechanism is proposed in [37] for the configuration of LTE transmission gaps, so that a fair coexistence, i.e., spectrum sharing, between LTE and Wi-Fi networks is guaranteed. Simulation results show that the proposed Q-Learning based approach improves the overall system capacity performance by 19 % and Wi-Fi capacity performance by 77 % when compared to a scenario with fixed duty cycles where highest aggregate capacity is achieved. The results show that the approach enables effective coexistence of LTE and Wi-Fi systems in the unlicensed spectrum bands.

In [38] the authors demonstrate that a Neural Network (NN) can accurately predict slots in a Multiple Frequencies Time Division Multiple Access (MF-TDMA) network. Through spectrum observation, the proposed Neural Network models are able to do online learning and predict the behavior of spectrum usage a second in advance. Results show that the proposed approach reduces the number of collisions by half when nodes follow a Poisson traffic distribution and a reduction by a factor of 15 when more periodic traffic patterns are used.

In [39] the authors propose a NN based approach that adapts LBT's CW size based on the predicted number of Negative Acknowledgments (NACKs) for all subframes in a Channel Occupancy Time (COT) of LTE-LAA. The correct configura-

tion of the CW size is of utmost importance to avoid collisions or to resolve contention among colliding radios. The proposed approach learns from previous experiences how many NACKs per subframe of a COT were received under certain channel conditions. After the learning phase, it is able to predict the number of NACKs for all subframes in a COT without having to wait (at least 4 ms) for delayed HARQ feedbacks. The CW size is exponentially increased upon the reception of a NACK for each subframe of a COT. Results show that the proposed approach provides better trade-off between fairness to Wi-Fi and LTE-LAA in terms of throughput and latency when compared to state-of-the-art approaches.

All the above mentioned works propose novel ideas to improve the spectrum utilization, however, all the results are simulation based. These ideas need to be evaluated to check their limits and capabilities of achieving improved spectrum utilization with real-world equipment. Therefore, the aim of our work is to fill this gap by providing a framework that can be used to implement these ideas, assess and validate them trough experimental validation.

III. PROPOSED FRAMEWORK

In this section the proposed framework is described in details. It is comprised of three main modules: (i) slot-

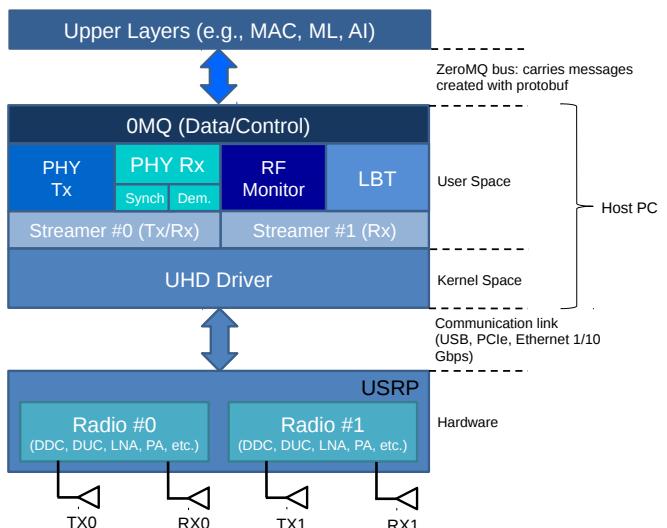



Fig. 2.   Proposed Framework Architecture.

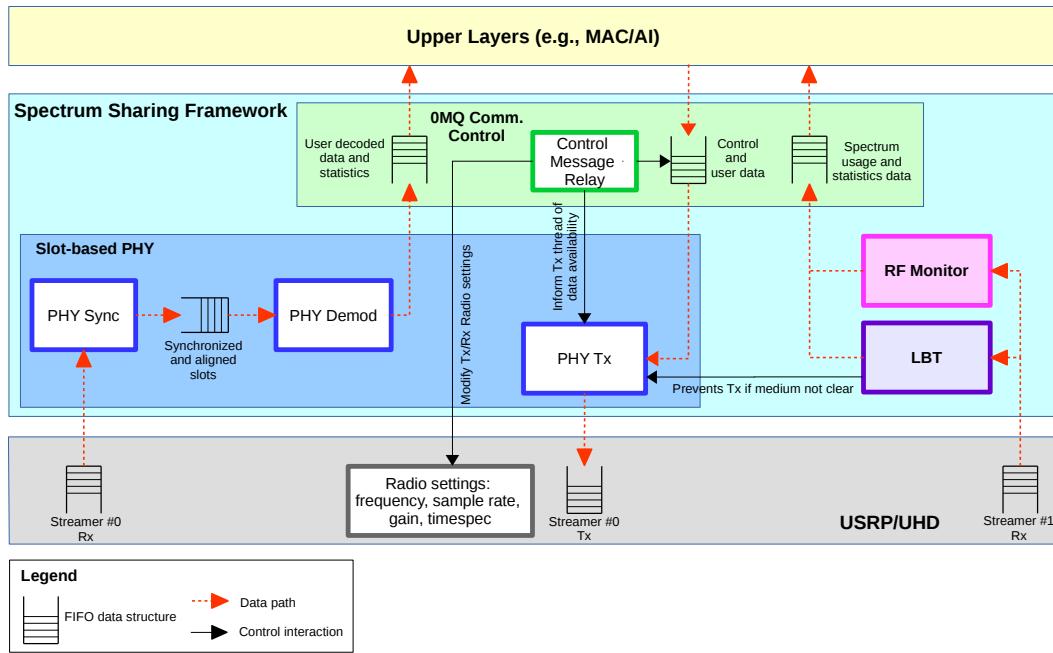



Fig. 3.   Threading architecture of the proposed spectrum sharing framework.

based PHY layer (for discontinuous data transmission and reception), (ii) RF Monitor (for spectrum sensing purposes) and (iii) Listen Before Talking (LBT) (for Radio Access Technology (RAT) coexistence). The high level architecture of the proposed framework is shown in Figure 2. All the main modules are connected to the ZeroMQ (Data/Control) module, also known as 0MQ, which interconnects the framework with upper layers through the ZeroMQ bus [40]. This module manages the exchange of control and statistics messages between the framework and upper layers. The framework is used in conjunction with Ettus USRP X family of SDR devices including NI's RIO platforms [41, 42] and communicates with it through the UHD driver and its APIs [26].

USRP devices are hardware platforms developed for software defined radio (SDR), and are commonly used by research labs, universities, and hobbyists. USRP devices are controlled through the UHD driver. The driver provides an Application Programming Interface (API) that enables controlling and accessing all features provided by USRP devices, while abstracting the low level implementation details of the hardware. The most important operations provided by the UHD driver fall into two classes, namely, control and streaming. Control operations are used to set frequency, sample rate, gain, etc. of the radios. Streaming operations are carried out through streamer objects, which allow applications running on the host to connect to Tx and Rx sample streamers offered by the USRP device. These samples come from the radios available in the devices. An Rx streamer object allows applications to receive samples from the device and a Tx streamer object allows applications to transmit samples to the device [26].

The slot-based PHY exclusively uses Streamer #0, both for Tx and Rx sample streams, and is fed by Radio #0. The

RF Monitor and LBT modules share Streamer #1 (both of them only use the Rx sample stream), which is fed by Radio #1. Streamers #0 and #1 can be configured independently, i.e., different TX/RX frequencies, gains and samples rates can be set for each of them independently. For performance and priority management reasons, each one of the modules (ZeroMQ Data/Control processing, PHY Tx, PHY Rx Synchronization, PHY Rx Data Demodulation, RF Monitor and LBT) runs on an exclusive thread. The reason why PHY and the sensing modules (i.e., RF Monitor and LBT) use different and independent Streamers is due to the fact that we want to decouple their operations. The RF Monitor and LBT modules use an exclusive streamer (i.e., Streamer #1) so that they can independently (from PHY RF parameters) and uninterruptedly (i.e., it is not desired to time-share a Streamer between PHY and the sensing modules) assess any configurable spectrum bandwidth at any desired center frequency and Rx gain. With this approach, both PHY and the sensing modules will always be receiving at their independently configured center frequencies, bandwidths and Rx gains without being affected by or affecting each other's operation. In the current implementation, only one of the two sensing modules, i.e., either RF Monitor or LBT, is started and connected to Streamer #1. The module to be started is configured through command line during initialization of the framework.

Communication with the proposed framework is entirely realized through a well-defined interface designed with Google's Protocol Buffers (protobuf) [43] for data serialization coupled with the ZeroMQ messaging library [40] for distributed exchange of control, statistics and data messages. Implementing the ZeroMQ push-pull pattern allows local or remote upper layers' real-time configuration of several parameters and

reading of several pieces of information/statistics provided by the framework. Based on the ZeroMQ logic, all modules are able to exchange control and data messages following a non-blocking communication paradigm. The proposed framework was designed to be totally decoupled and independent of upper layers modules (i.e., MAC, intelligence, etc.), not posing any constraints on hardware and software adopted by them. This way, upper layer modules can employ any type of hardware (e.g., GPUs, FPGA, etc.) or software algorithms in their implementation. Additionally, by adopting protobuf and ZeroMQ, the proposed framework offers other modules/layers the flexibility to use any kind of programming language implementation, making it easier to integrate multiple independent modules into a single working system. The flexible communication interface provided by the framework makes it ideal for deployment in testbeds and allows different upper layers to be plugged in, hence, an important enabler for intelligent spectrum sharing experimentation towards next generation of wireless networks.

Figure 3 illustrates the different layers composing the proposed spectrum sharing framework and the threads within each one of them. Red dashed arrows indicate data paths while black arrows indicate control/information interaction between threads.

Through the use of the proposed framework, a myriad of real-world experiments can be performed, ranging from research of new physical layer techniques (e.g., higher order modulation schemes, novel waveforms and numerologies, channel coders, etc.) to RAT coexistence in unlicensed and licensed-shared spectrum bands (e.g., comparison of the fairness between different contention-based mechanisms such as Carrier Sense Adaptive Transmission (CSAT) and LBT, impact of different parameters such as COT, idle time, Tx power, PHY bandwidth, spectrum overlap, Modulation Code Scheme (MCS) on the performance of Wi-Fi and LTE networks, etc.) to collaborative and intelligent radio networks (i.e., networks where AI and ML components act as the brain of the system and control various aspects of medium access and PHY layers to ensure adaptability, while also running some sort of collaboration protocol [44, 45]).

In the next sections, we describe each one of the main modules present in the proposed framework.

IV. Slot-based Physical Layer

In spectrum sharing scenarios, several networks might share the medium at the same time and therefore, continuous access to it may not be possible all the time. With discontinuous transmissions, it is possible to have a better use of the available spectrum band and to coordinate its usage with other networks in an opportunistic/intelligent/collaborative way. Additionally, continuous access is the preferred approach for systems operating in exclusive licensed bands where there is no need for sharing, and is not really suited to fair medium sharing between co-located networks. Moreover, in unlicensed spectrum, some regulatory authorities do not allow continuous transmissions and limit the maximum duration of a transmission burst [46].

A PHY layer supporting a discontinuous transmission feature would have a great value to spectrum sharing research, as it could be used to leverage techniques devised to achieve fairer coexistence and higher spectral efficiency in scenarios where time-frequency resources are shared. Hence, as part of our framework for spectrum sharing research, we decided to develop a discontinuous transmission-based PHY, which transmits data bursts in small transport units called time slots, or slots for short. A slot is the container through which data is exchanged in the network.

We have based the development of the proposed slot-based PHY on the LTE PHY standard, as it offers several advanced features including high spectrum efficiency, multiple bandwidths, high peak data rates, mobility, multi-user access, flexible time framing and time-frequency structure, link adaptation with adaptive modulation and coding schemes and multiple-input and multiple-output (MIMO) [47]–[49]. Moreover, the next generation of wireless networks (i.e., 5G) will naturally evolve from LTE -based standards [47], i.e., they will be made available through improvements in LTE, LTE-Advanced and LTE-Advanced Pro technologies [50]. Consequently, building upon the LTE PHY standard makes it easier to further extend the slot-based PHY to be compliant with 5G standards and further evolutions. The proposed slot-based PHY is built upon the srsLTE library [29], and therefore, absorbs and evolves on top of the existing LTE features. srsLTE is a free and open-source LTE software library developed by Software Radio Systems (SRS) [29].

In order to stay aligned with both LTE standards and 5G initiatives [49, 50], we decided to adopt an Orthogonal Frequency-Division Multiplexing (OFDM) based waveform. OFDM is a mature technology, which is vastly implemented in a great number of products due to its several advantages such as robustness to severe multipath fading, low implementation complexity, easy integration with MIMO, simple channel estimation, etc. [49]. Additionally, OFDM allows for enhancements such as waveform windowing/filtering, which can effectively minimize out-of-band spurious emissions [51]. It is also worth mentioning, that compared to other popular OFDM-based technologies like Wi-Fi [52] and Digital Video Broadcasting-Terrestrial (DVB-T) [53], LTE was devised, since its inception, for multi-user communications by assigning subsets of 12 subcarriers, also known as a physical resource block (PRB) or resource block (RB) for short, over multiple 1 ms long subframes to individual users. This flavor of multiple access is known as Orthogonal Frequency-Division Multiple Access (OFDMA) and gives LTE a very flexible way of allocating time/frequency resources to multiple concurrent users.

As mentioned earlier, the proposed slot-based PHY module is split into three submodules, namely, PHY Tx, PHY Rx Synchronization and PHY Rx Demodulation where each one of them runs on an exclusive, standalone thread. The reason for having a multi-threaded PHY implementation is that it allows independent critical and/or time-consuming tasks to be executed simultaneously (i.e., concurrently). Computing

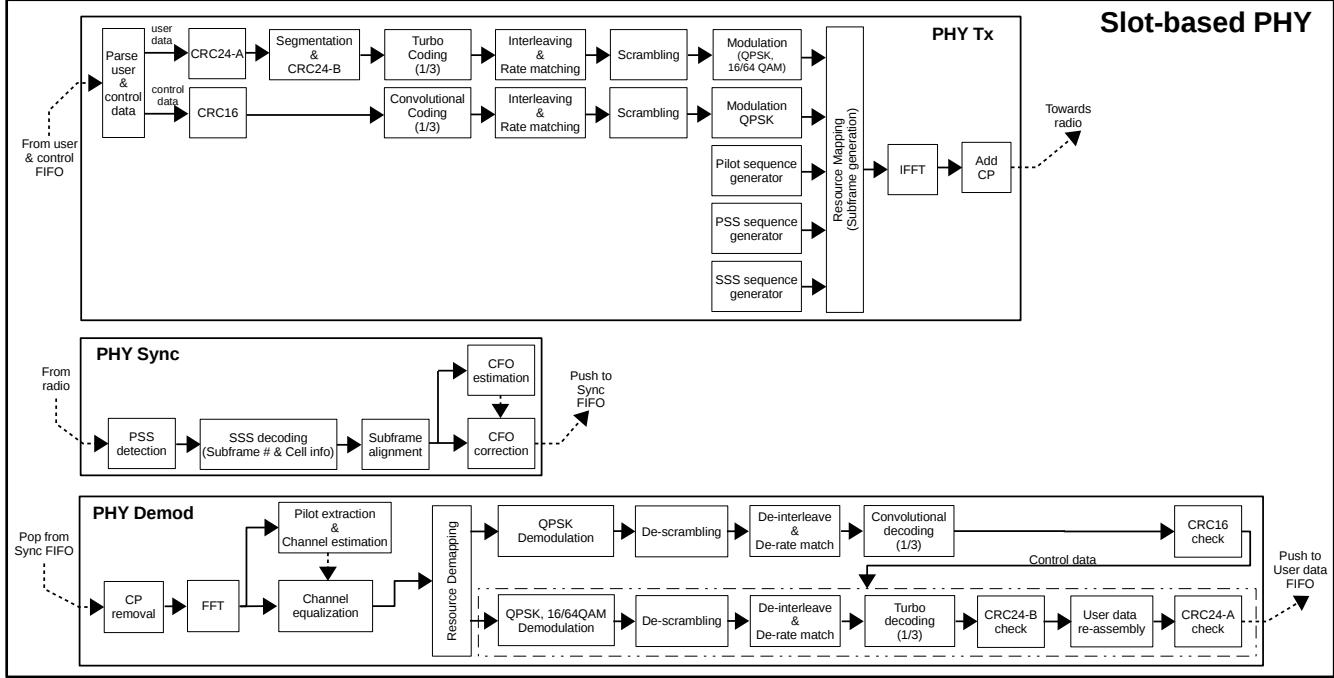



Fig. 4.   Block diagram of the slot-based PHY transmitter and receiver sides.

performance and efficiency is improved by taking advantage of concurrency. Allied with multi-core enabled Central Processing Units (CPUs), the multi-threaded PHY naturally supports full-duplex communications mode, i.e., one PHY can simultaneously transmit and receive at different frequencies, which consequently results in a higher throughput. PHY Tx thread is responsible for modulation and transmission of data (i.e., user and control data). PHY Rx Synchronization thread is responsible for the Primary Synchronization Signal (PSS) and Secondary Synchronization Signal (SSS) detection, Carrier Frequency Offset (CFO) estimation/correction and slot (subframe) time-alignment tasks. CFO is estimated based on the PSS signal [54]. PHY Rx Demodulation thread takes care of user and control data demodulation, i.e, OFDM demodulation (FFT processing and CP removal), channel estimation/equalization, resource demapping, Physical Downlink Control Channel (PDCCH) decoding and Physical Downlink Shared Channel (PDSCH) decoding. The slot-based PHY receives data and control messages from the ZeroMQ Data/Control module. Decoded user data and statics related to the PHY operation (Rx/Tx statistics) are sent directly to the upper layers through 0MQ bus. Figure 4 depicts block diagrams of the slot-based PHY transmitter (PHY Tx thread) and receiver sides (PHY Sync and PHY Demod threads).

Regarding numerology, the proposed PHY supports LTE numerology with the same subcarrier spacing (15 kHz) and the cyclic prefix (CP) lengths (normal and extended CPs). Different CP lengths are used to accommodate different levels of inter-symbol interference (ISI) at different frequencies, coverage and mobility scenarios [50]. However, it also supports the configuration of different (larger and smaller) subcarrier

spacings and CP lengths by adopting different sampling rates. Larger subcarrier spacing can be used to (i) decrease latency due to resulting smaller subframe durations, which is very important for URLLC applications requiring very low latencies, (ii) mitigate inter-carrier interference (ICI) in V2x applications at high frequency bands (due to Doppler shift caused by high-speed scenarios) [48]. On the other hand, smaller subcarrier spacing is interesting for (i) extended coverage, due to increased Power Spectral Density (PSD), and (ii) high delay spread scenarios (due to longer CP) at low frequency bands [49, 50].

All LTE predefined bandwidths (1.4, 3, 5, 10, 15 and 20 MHz) are supported. PHY channel bandwidth (BW) can be changed through command line at start up, or in real-time, through the Tx control message, which is introduced later in this section.

Differently from the LTE PHY downlink, which is continuously transmitting signals, the proposed slot-based PHY employs discontinuous (i.e., bursty) transmission of slots. A (time)-slot is the basic transmission unit of the proposed PHY and each one is 1 ms long. Depending on its type, a slot might carry synchronization signals, reference signals, broadcast, control and user data.

Three different types of slots, following the same definition as the LTE subframes, are proposed [47, 49]. All the three slot types for the normal CP length are depicted in Figure 5. The first type, which is equivalent to subframe #0 in the LTE standard, carries synchronization signals (PSS/SSS), reference signals, broadcast, user and control data. The second type is equivalent to LTE's subframe #5 and carries synchronization signals (PSS/SSS), reference signals, user and control data.

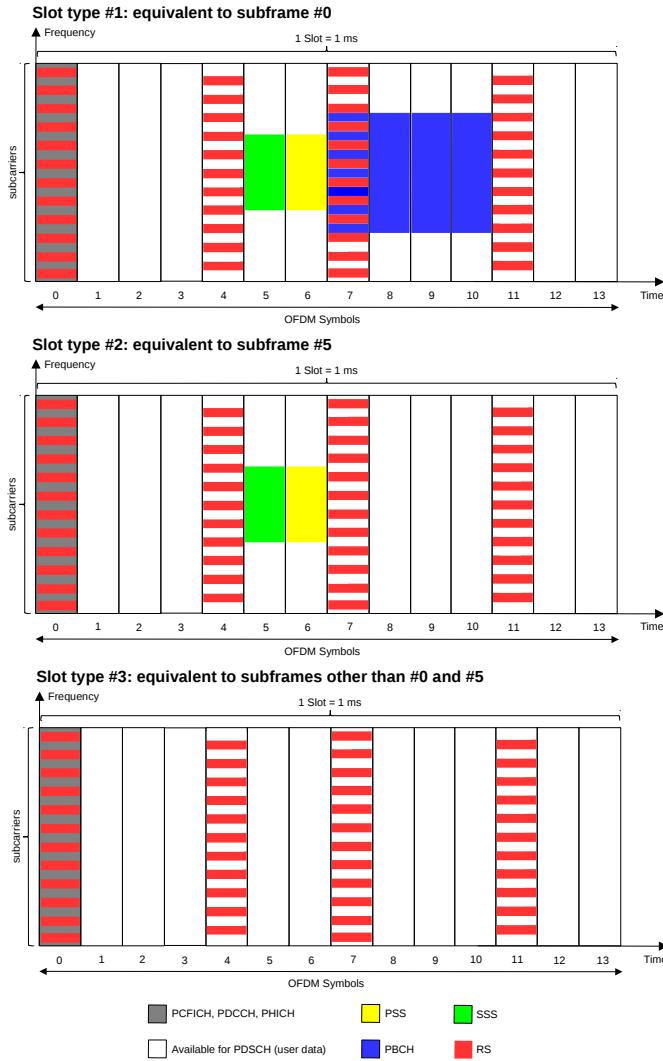



Fig. 5.   Proposed slot types for normal CP length.

The third type, which is equivalent to LTE's subframes other than #0 and #5 (i.e., #1 to #4 or #6 to #9), only carries reference signals, user and control data. It is important to notice that the RBs intended to carry user data can be split into different sections and allocated to different concurrent users. All slot types carry user data, signals for channel estimation/equalization, and control data, which holds information on how to decode each one of the possible user data sections. The first two slot types are used for transmission detection, as they carry synchronization signals. The first slot type is used whenever there is a need to transmit broadcast information. As it does not carry synchronization signals, the third slot type can only be used if it is transmitted after one of the other two types. Similarly to an LTE subframe, each slot can have 14 or 12 OFDM symbols depending on the configured CP length, i.e., normal or extended CP [47]. Moreover, each one of the three slot types is self-decodable, i.e., they always carry control data necessary for the receiver to correctly decode the user data section. The control data part of the slot is used at

the PHY receiver side to automatically detect the number of allocated RBs, the location of the allocated RBs in the resource grid, and the MCS used to transmit data of a specific user. By following this approach, upper layers do not need to specify the number of allocated RBs, RB location and MCS before every slot reception. Upon correct user data reception, PHY informs to the upper layers the number of received bytes and the corresponding MCS.

The proposed PHY allows bursty transmissions with variable COT, i.e., the number of slots to be transmitted in a row without any gap (i.e., idle time) between them is variable. The number of slots in a COT, i.e., frame, is derived based on MCS, number of RBs and data length (i.e., number of bits to be transmitted) parameters sent by upper layers in the control message. The minimum COT is equal to 1 ms and is equivalent to a slot. Variable COT enables the support of different traffic loads and channel occupancy. Every slot can carry a pre-defined number of bits, which is based on the MCS and number of RBs used for a transmission [49].

The proposed slot-based PHY can operate with two different data transmission modes. The first mode can be used to establish communications with multiple nodes in ad-hoc or mesh networks, depending on routing capabilities implemented in the upper layers connected to the PHY (i.e., routing at layer 3 of the OSI model). This mode is a simplification of the LTE channel's structure and uses the Physical Downlink Shared Channel (PDSCH) for user data transmissions in both directions, i.e., downlink and uplink, in order to take advantage of both multi-user communications and high performance Forward Error Correction (FEC) provided by turbo codes [49]. Additionally, this mode also uses Physical Control Format Indicator Channel (PCFICH) and Physical Downlink Control Channel (PDCCH) for control purposes and Cell-specific Reference Signals (CRS) for channel estimation and equalization. The second mode operates exactly with the same channel's structure as adopted in the LTE standard, i.e., it has all downlink and uplink physical layer channels as defined in the standard. This mode can be used to establish communications the same way as in a cellular mobile network. In both modes, OFDMA can be used to establish communications with multiple users at the same time.

Regarding frame structures, we propose two different types, as depicted in Figure 6. In the first one, PSS and SSS sequences are added to all slots in a COT, allowing for better time-frequency tracking at the receiver side. This frame structure allows the receiver to easily perform slot synchronization every 1 ms without the necessity of complex tracking algorithms. A downside of this structure type is that PSS and SSS detection is time-consuming and consequently, decreases the maximum achievable throughput as the period of time required to detect and decode slots increases. In the second proposed frame structure, PSS and SSS signals are added only to the very first slot of a COT. This frame structure improves the throughput as PHY only needs to synchronize once (i.e., detect PSS, decode SSS, time-align and estimate/correct CFO), consequently, decreasing the overall slot decoding time

Standard LTE Frame Structure

Proposed Frame Structures

1) Add PSS/SSS to all slots.

2) Add PSS/SS to first slot only.

*M depends on # RBs, MCS and data length.

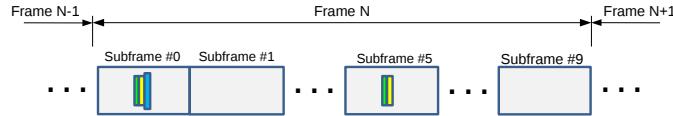
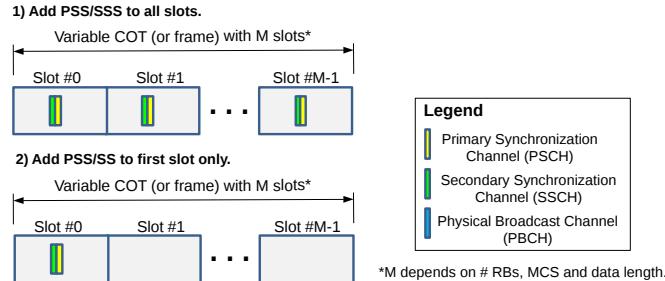



Fig. 6.   Standard LTE versus Proposed Frame Structures.

[55]. However, this frame structure requires a more complex tracking/synchronization algorithm and the necessity to add information about the number of slots in a COT to the first slot, which means, if the first slot is lost, none of the subsequent slots can be decoded. Moreover, if the CFO is wrongly estimated or changes throughout the duration of a COT (i.e., frame), then the slots might be incorrectly decoded.

The first proposed frame structure is more robust to time and frequency fluctuations and should be used in cases where transmitter and receiver nodes are moving as it provides better synchronization and less slot losses. On the other hand, the second frame structure should be used for static or quasi-static nodes. The proposed PHY allows switching between the two frame structures either through command line at initialization or during run-time in order to adapt its operation to different radio link conditions.

The Payload Data Unit (PDU) adopted by the proposed PHY is a Transport Block (TB), which is a concept reused from LTE PHY layer. Therefore, a TB is the payload coming from upper layers and given to PHY to be transmitted over the air through the PDSCH. A TB is defined in the LTE standard and varies according to some parameters [56]. According to the LTE standard, 1 TB consists of a number of bits that can be accommodated within a 1 ms long subframe given the selected number of RBs and MCS [56]. Therefore, given the number of allocated RBs and the desired MCS, upper layers can find the number of bits that can be handled by an 1 ms long slot.

The communication between upper layers, e.g., Medium Access Control (MAC), and the proposed PHY layer is carried out through the exchange of four messages. The first two, namely, Tx and Rx Control messages, are used to manage slot transmission and reception respectively. The parameters carried by these two messages can be configured and sent to PHY by upper layers before the transmission of every slot, hence allowing runtime configuration. The other two messages, namely, Tx and Rx statistics messages, are used to provide real-time feedback from PHY to upper layers, yielding

vital information necessary for such layers to take actions.

Tx control messages carry the user data (i.e., TB) to be transmitted and Tx parameters related to that transmission, namely, number of RBs, MCS, data length, Tx gain, Tx channel, Tx PHY BW, transmission timestamp and frame structure type. The transmission timestamp parameter enables time-scheduled transmissions, which allows the implementation of Time Division Duplexing (TDD) or Time Division Multiple Access (TDMA) technologies with the proposed slot-based PHY layer. **Rx control** messages are used to configure Rx channel, Rx gain and Rx PHY BW.

The other two messages, namely, Rx and Tx statistics, are used to inform upper layers of PHY Rx and Tx processing results respectively. **Rx statistics** messages carry the received data and reception statistics related to the received data such as Channel Quality Indicator (CQI), Received Signal Strength Indication (RSSI), decoded MCS, slot error counter, decoding time, etc. **Tx statistics** messages inform upper layers of transmission statistics like coding time, total number of transmitted slots. Table II summarizes all the real-time configurable parameters and statistics offered by the slot-based PHY.

Figure 7 depicts the GUI provided by the slot-based PHY, which can be used for debugging or demonstration purposes. The GUI plots the constellation diagram of equalized and decoded PDSCH and PDCCH symbols (upper and lower left respectively), the equalized channel response (upper right) and the correlation peak of the PSS (lower right).

[1]Depends on the USRP daughter board installed [26].

[2]The numbers correspond LTE bandwidths: 1.4, 3, 5, 10, 15 and 20 MHz respectively.

[3]The numbers correspond to frame structure types where PSS/SSS signals are added to all slots in a burst and PSS/SSS signals are only added to the first slot in a burst respectively.

TABLE II

SLOT-BASED PHY REAL-TIME CONFIGURABLE PARAMETERS AND STATISTICS.

Message	Parameter	Type	Unit	Range
Tx control	Number of RBs	uint32	-	1-100
	MCS	uint8	-	0-28
	Tx gain	uint32	dB	depends on HW [1]
	Tx channel	uint32	-	≥ 0
	Tx PHY BW	uint8	MHz	0-6 [2]
	Transmission timestamp	uint64	s	≥ 0
	Frame structure type	uint8	-	0/1 [3]
	Data length	uint32	-	> 0
	User data	uchar[]	-	uchar range
Rx control	Rx channel	uint32	-	≥ 0
	Rx gain	uint32	dB	depends on HW [1]
	Rx PHY BW	uint8	MHz	0-6 [2]
Rx statistics	CQI	uint8	-	0-15
	RSSI	float	dBW	float range
	Noise	float	dBW	float range
	Decoded MCS	uint8	-	0-28
	Slot error counter	uint32	-	≥ 0
	Decoding time	uint32	ms	≥ 0
	Data length	uint32	-	≥ 0
	Received data	uchar[]	-	uchar range
Tx statistics	Coding time	uint32	ms	≥ 0
	Number of transmitted slots	uint32	-	≥ 0

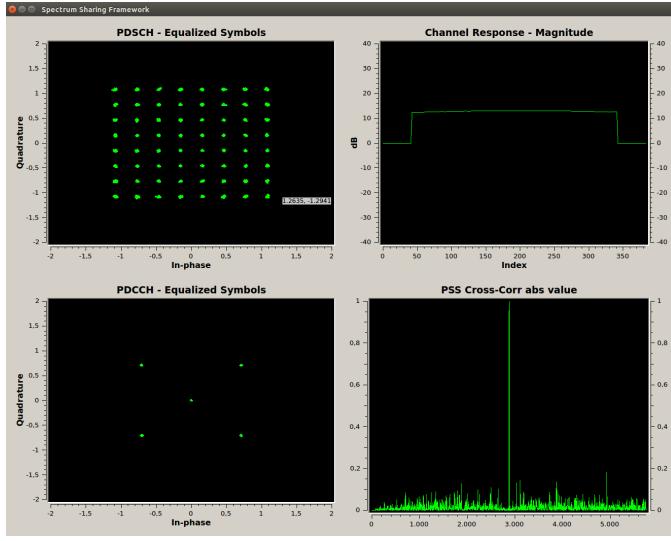



Fig. 7.   Slot-based PHY graphical user interface.

A. Filtered-OFDM

As it is well-known, OFMD-based waveforms are not suited for spectral coexistence due to their poor spectral localization [57]. This problem is caused by the rectangular pulse-shape used in OFDM, which leads to a sync-pulse property in the frequency domain with a very low second lobe attenuation of -13 dB [58].

One of the simplest but still very efficient approaches used to guarantee better spectral localization, i.e., lower out-of-band (OOB) emissions, and maintain the complex-domain orthogonality of the OFDM symbols is to apply some sort of filtering to the time domain OFDM symbols, giving rise to a new waveform known as Filtered-OFDM (f-OFDM) [59].

Appropriate filtering of the OFDM signal must satisfy the following criteria: (i) have a flat passband over the subcarriers in the sub-band; (ii) have a sharp transition band to minimize guard-bands; (iii) have sufficient stop-band attenuation. A Finite Impulse Response (FIR) filter with a rectangular frequency response, i.e., a sinc-pulse impulse response, meets these criteria. To make this causal, the low-pass filter is realized using a window, which, effectively truncates the impulse response and offers smooth transitions to zero on both ends of it [60]. This filter causes no distortion in the pass-band and exhibits total out-of-band rejection. Therefore, the filter used in this work is defined in time domain as

$$f(n) = p(n).w(n), \tag{1}$$

where $p(n)$ is the sinc impulse response and $w(n)$ is the truncation window. These two functions are respectively defined as

$$p(n) = \begin{cases} \sin\left(\frac{12\pi \text{NPRB} n}{\text{NFFT}}\right) / \frac{12\pi \text{NPRB} n}{\text{NFFT}}, & n \neq 0, \\ 1, & n = 0. \end{cases} \tag{2}$$

$$w(n) = \left\{ \frac{1}{2}\left[1 + \cos\left(\frac{2\pi n}{L-1}\right)\right] \right\}^{0.6}, \tag{3}$$

where NPRB is the number of physical resource blocks, NFFT is the FFT length used in the OFDM modulation, L is the filter length and $-\frac{(L-1)}{2} \leq n \leq \frac{(L-1)}{2}$.

The window defined in (3) exhibits smooth transitions to zero on both of it ends. These smooth shifts are very import to avoid abrupt transitions at the beginning and end of the truncated filter, $f(n)$, which in consequence, refrains the truncated filter from spilling over adjacent frequency. A very good time/frequency localization is conferred to the impulse response of the truncated filter by applying the window, $w(n)$, to the Sinc function. Additionally, the windowing process keeps the ISI resulting of the f-OFDM signal modulation within acceptable levels once the main filter's energy stays restricted to its main lobe in time domain [60].

As showed in Figure 8, the slot-based PHY creates its filtered OFDM signal based on the modulation of M consecutive subcarriers in a number of consecutive OFDM symbols. The OFDM subcarriers are modulated with synchronization, reference and user/control data symbols (e.g., PSS/SSS sequences, QPSK/16QAM/64QAM and Reference symbols, etc.). During each OFDM symbol period, the PHY generates a N-point inverse fast Fourier transform (IFFT) out of the M input symbols (i.e., the actual/useful OFDM symbol) and adds a CP to the front of the OFDM symbol. Note that N is the FFT size of the system, which is defined by the sampling rate and subcarrier frequency spacing, and that $N > M$. The unused OFDM subcarriers, $N - M$, are set to zero (i.e., *nulled*). The null subcarriers have two purposes: (i) the ones along both edges of the spectrum are used as guard-bands between adjacent channels, (ii) the one at Direct Current (DC), i.e., the central subcarrier, is used to allow the use of simple and cheap direct-conversion (zero intermediate frequency) receivers. Therefore, mathematically, the output signal of the slot-based PHY transmitter can be expressed as

$$s(n) = \sum_{l=0}^{L-1} s_l(n - l(N + N_g)), \tag{4}$$

with

$$s_l(n) \triangleq \sum_{m=m'}^{m'+M-1} d_{l,m} e^{j2\pi mn/N}, \quad -N_g \leq n < N, \tag{5}$$

where N_g is the length of CP, $d_{l,m}$ is the symbol modulating m-th OFDM subcarrier of the l-th OFDM symbol, L denotes the number of OFDM symbols making up a subframe and $\{m', m'+1, \cdots, m'+M-1\}$ is the assigned subcarrier range. The filtered OFDM signal is then obtained by passing the signal $s(n)$ through the designed spectrum shaping filter, $f(n)$, defined in (1), and defined as

$$\tilde{s}(n) = s(n) * f(n). \tag{6}$$

The spectrum shaping filter $f(n)$ has pass-band width equal to the total frequency width of the assigned subcarriers, i.e., the number of allocated RBs, and its time duration is only a fraction of the duration of an OFDM symbol. The filter

duration is chosen in order to maintain OFDM's good time-localization feature [48].

Therefore, a FIR filter meeting the necessary requirements is added to the Tx processing flow, just after the OFDM modulation. The filter is applied just before the OFDM symbols are sent to the hardware, e.g., USRP. An efficient FIR is implemented on software making use of Single Instruction Multiple Data (SIMD) instructions present on most of the general purpose microprocessors, i.e., central processing units (CPUs), available today to exploit the data-level parallelism present in the filtering processing. In Appendix I, we present an example of an optimized digital FIR filter implementation using Advanced Vector Extension (AVX) instructions. Our comparisons with a naive implementation show that such SIMD instructions substantially speed-up the filtering processing. The filter's coefficients applied to the modulated OFDM signal are automatically selected according to the configured Tx PHY BW (i.e., the coefficients are selected in real-time based on the Tx PHY BW field in the Tx control message) as it needs to have its cut-off frequency changed to exactly filter the desired signal's bandwidth. Figure 8 shows the f-OFDM enabled transmitter scheme, which can have the FIR filtering processing enabled or disabled and the filter order selected at command line during start up of the framework.

B. Advantages of the proposed slot-based PHY

In this section we describe the main advantages offered by the proposed slot-based PHY. The first advantage is that there is no need to align the start of transmissions to subframe boundaries. This is in contrast to LTE-LAA, where reservation signals are transmitted in order to prevent other RATs (e.g., Wi-Fi) from initiating a transmission until the next subframe boundary. The transmission of reservation signals clearly reduces the utilization efficiency of the available radio resources. As the proposed PHY adopts bursty transmissions, all the information required for synchronization and demodulation is self-contained in every slot. This allows for the development of standalone solutions that do not need to rely on reservation signals nor be subframe-aligned to an anchor on licensed spectrum.

A second advantage of the proposed PHY is that the filtering applied to the OFDM symbols makes the filtered-OFDM PHY more spectral efficient as the OOB emissions are reduced. To be more specific, in LTE, 10% of the allocated bandwidth is reserved as guard band, which allows the waveform signal to attenuate and therefore, meet the spectrum mask requirements [61]. Undoubtedly, this is a considerable

Fig. 9.   Closer coexistence to other RATs due to reduced OOB emissions.

waste of frequency resources, which are becoming more and more precious. Additionally, OOB emissions interfere with other closed (i.e., nearby) allocated systems, decreasing the quality of the received signal, which impacts on the throughput experienced by that system. The reduced OOB emissions make the proposed PHY ideal for coexistence with other RATs (e.g., incumbent systems, spectrum sharing RATs, etc.), allowing it to operate closer to other systems in frequency domain and consequently, reducing spectrum wastage while increasing the spectral efficiency as shown in Figure 9.

Another advantage, offered by the proposed framework, is the possibility to configure in real-time all PHY (and also RF Monitor and LBT modules) parameters through pre-defined control messages, which can also be easily modified (thanks to the use of Google's protobuf interface description language) to accommodate new parameters. By plugging in a new module to the system, it is possible to experiment with the PHY parameters in a programmatic way, so that new algorithms can be developed and tested.

V.   RF MONITOR

As mentioned earlier, this module runs on an exclusive thread and utilizes Radio # 1 of the USRP to act as a spectrum

Fig. 8.   Filtered-OFDM enabled transmitter.

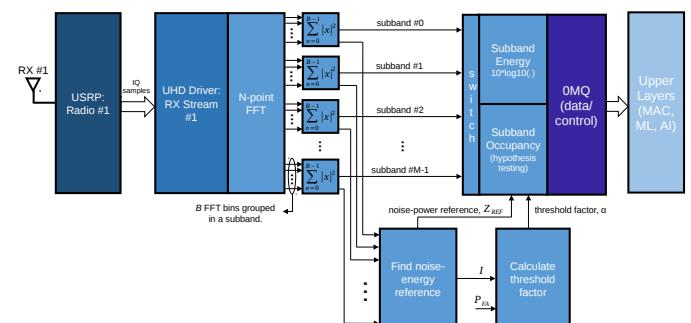



Fig. 10.   RF Monitor architecture.

TABLE III

RF Monitor real-time configurable parameters and statistics.

Message	Parameter	Type	Unit	Range
RF Monitor control	Monitored bandwidth	float	Hz	depends on HW [1]
	Center frequency	float	Hz	depends on HW [1]
	Rx gain	uint32	dB	depends on HW [1]
	Number of FFT bins	uint32	-	16-16384
	Report periodicity	uint32	μs	≥ 150
RF Monitor statistics	IQ samples timestamp	uint64	s	≥ 0
	Power/Occupancy array	float[]/bool[]	-	float range or True/False

sensing monitor. It monitors and reports on a defined range of spectrum band and can be configured to report either the detected power per subband or subband occupancy (statistical hypothesis testing [62]), i.e., if a subband is free or already being occupied by another transmitter. Figure 10 depicts the architecture of the RF Monitor module.

Subband power reporting is obtained by applying a N-point FFT to the received IQ samples, summing the squared modulus of every consecutive B FFT bins (where B must be a power of two) to form M groups of power samples, i.e., subbands and then, applying the base-10 logarithm in relation to a reference power of 1 Watt to each one of the subbands, resulting in measurements given in dBW. A M-long float array holds the subband power information.

In order to report subband occupancy, the module implements a spectrum-sensing algorithm based on a Cell-Average Constant False Alarm Rate (CA-CFAR) strategy [62, 63]. The proposed algorithm is composed of three stages: (i) identification of a set of power samples, I, that can be considered as containing only the presence of noise in order to calculate a noise-power reference value, Z_{REF}; (ii) the number of power sample values considered for the calculation of the noise-power reference value, and the desired probability of false alarm, P_{FA}, are used in the second stage to calculate the detection factor, α; (iii) the third stage employs both, the threshold factor and the noise power reference value to test the occupancy of each one of the subbands.

The power samples used in the first stage are obtained after applying an N-point FFT to the received IQ samples and summing the squared modulus of every B FFT bins into subbands. The procedure used to find the noise-power reference value, Z_{REF}, in the first stage involves ordering the subbands in ascending order of power. The detection threshold is found by multiplying Z_{REF} by the threshold factor, α. In the end, there are M subbands which will be tested against the detection threshold, in order to declare whether the subband is occupied or free. If the subband power is greater than the detection threshold, then the subband is declared occupied (true), otherwise it is declared free (false). After testing all subbands, an M-long boolean array holds the occupancy information of the monitored spectrum band. In the current implementation, the probability of false alarm, P_{FA} is set to 10^{-4}. Further information on the subband occupancy algorithm is provided in Appendix II.

The module allows configuration of the following parameters for both types of reporting (power or occupancy): monitored bandwidth (through the change of the sampling rate),

center frequency, number of FFT bins M, which changes the frequency resolution, Rx gain, the number of consecutive FFT bins, B, considered as a subband (i.e., number of bins considered for the subband power calculation) and the periodicity of the statistics report. These parameters can be configured in real-time by upper layers through the **RF Monitor control** message. The sensing reporting type, i.e., power or occupancy, is configured during initialization of the framework. The following information is sent to upper layers through an **RF Monitor statistics** message: timestamp of the moment the IQ samples were received by the USRP and power or occupancy array depending on the configured reporting type. Table III summarizes all the real-time configurable parameters and statistics offered by the RF monitor module.

The RF monitor module is of great importance to spectrum sharing mechanisms as it offers a local insight of the spectrum band usage. It allows upper layers to access spectrum sensing measurements, which can be used to train ML and AI modules to better understand the environment, optimize the spectrum usage/sharing and cooperatively work with other networks without any previous knowledge on the other network's operation and implementation, i.e., without any co-design [64]. For instance, this module allows the implementation of adaptive carrier selection algorithms such as CSAT [65] and can also be used to train ML or AI modules to perform MF-TDMA slot allocation [38].

VI. Listen Before Talk

As the other modules in the proposed framework, the software-based Listen Before Talk (SW-based LBT) module also runs on an exclusive thread. The module implements a contention-based mechanism for medium access based on 3GPP's LAA specifications [66, 67]. However, differently from 3GPP's LAA specifications, the proposed implementation allows the following configurations: minimum and maximum contention window (CW) sizes (with the option for the upper limit to be fixed or dynamically adjusted), variable frame period (with configurable maximum COT and clear periods, which allows longer than the standardized maximum frame period [66] and the adaptation to different traffic loads and channel usage [68]) and threshold. Figure 11 shows the proposed variable frame structure supported by the implemented LBT module. The frame period, which is composed of the COT and clear periods, can be dynamically configured to any value greater than 1 ms.

The rationale behind the decision to implement a SW-based LBT module is explained next. We start by presenting in Table IV round trip time (RTT) measurements for an x310 USRP device combined with different communication links. Notice

TABLE IV

Round Trip Time between UHD and x310 USRP device [69, 70].

Link Type	RTT [us]	Host PC configuration
PCIe	79	Intel i7-6700 3.4GHz, NI PCIe x4 card
10 Gbps ETH	106	Intel E5-2650 v4 2.2GH, Qlogic 57810 ETH
1 Gbps ETH	101	Intel i7-6700 3.4GHz, Intel i219-v ETH

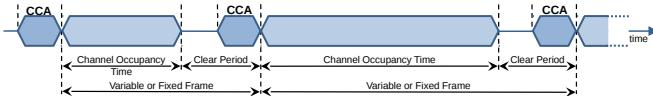
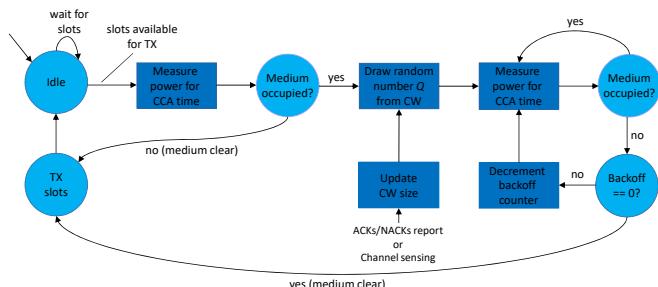



Fig. 11.   SW-based LBT frame structure.

that, each latency measurement is calculated as the average over 10 trials for each one of the following sampling rates: 5, 10 and 25 MHz. The measurements show that compared to the 1 ms long slot of the proposed PHY, it is feasible to use a SW-based LBT implementation as the RTT is of approximately 100 [us]. The SW-based implementation avoids the development an FPGA-based LBT module, which allows the framework to be used and ported to a wider range of SDR devices. Additionally, the SW-based implementation makes the LBT module easier to be customized by a wider range of users, as FPGA development is not so widespread and straightforward as software development.

Figure 12 depicts the Finite State Machine (FSM) of the proposed SW-based LBT module, which is explained as follows. Initially, a radio having data to transmit performs a Clear Channel Assessment (CCA) in order to determine if the medium is occupied or clear, i.e., the radio performs energy detection (ED) in given spectrum band during CCA observation time, if the measured energy level exceeds the CCA threshold then the medium is declared as occupied, otherwise, the medium is clear. If the radio finds the medium clear, it may transmit immediately and occupy the medium for the total time of COT.

In case the medium is occupied, the radio must execute a random backoff procedure, where a backoff counter is set to a random number, Q, drawn from a specified interval called the CW [71]. During the random backoff procedure, Q CCA checks are performed for the duration of a CCA observation time, where Q defines the total number of clear periods that need to be observed (i.e., counted) before the radio can transmit. The backoff counter is decremented every time the medium is observed to be clear. The radio may transmit when the counter reaches zero. The random backoff procedure was designed to cope with situations where more than one radio senses the medium as being in the clear state (i.e., there is no transmission occurring at that time) at the same time, and with this, decreasing the collision probability.

The CW size can be configured to be fixed or dynamic, with adjustable minimum and maximum values. In the fixed mode, Q is always drawn from the same interval, giving different radios the same probability of transmission opportunity in spite of collisions and traffic load. This approach is not fair when coexisting with RATs that adopt exponentially increasing CWs (e.g., Wi-Fi) when a collision happens. In the dynamic mode, a binary exponential backoff mechanism is used in the way that the CW size is increased exponentially based upon the occurrence of a collision (which can be based on feedback reports such as HARQ ACK/NACK or based on channel utilization sensing [72]) and reset to the minimum value when the transmission succeeds. The dynamic mode was devised to improve the overall channel utilization, reduce the collision probability and fairly coexist with other RATs like Wi-Fi.

The following parameters can be configured in real-time by upper layers through an **LBT control** message: sampling rate, center frequency, CCA threshold, CCA observation time, CW type (fixed or dynamic), CW minimum and maximum values and idle time. Additionally, the proposed LBT design keeps track of the following channels' occupancy statistics: channel clear/occupied ratios and clear/occupied average energies. These statistics are reported to upper layers through an **LBT statistics** message and can be used, for example, to find the optimum channel to be used, i.e., the least occupied channel. Table V summarizes all the real-time configurable parameters and statistics offered by the LBT module.

VII. USE CASES

In this section we describe three use cases for the proposed framework. The first one focuses on how the framework may facilitate slot selection in time and frequency domains in order to coexist with other technologies, based on intelligent prediction of the medium usage. The second use case shows how the framework can allow contention-based parameters at MAC level to be optimized when an intelligent upper layer is integrated for specific technologies. The third one shows how the framework can be used to devise and test flexible spectrum sharing paradigms.

A. Transmission Pattern Prediction

In the unlicensed 5 GHz spectrum band, RATs such as LTE-U, LTE-LAA and Wi-Fi use the same spectrum for

Fig. 12.   Software-based LBT FSM.

TABLE V

LBT REAL-TIME CONFIGURABLE PARAMETERS AND STATISTICS.

Message	Parameter	Type	Unit	Range
LBT control	Sampling rate	float	Sps	depends on HW [1]
	Center frequency	float	Hz	depends on HW [1]
	CCA threshold	float	dBW	float range
	CCA observation time	uint32	μs	> 0
	CW type	uint8	-	0/1 (fixed/dynamic)
	Minimum CW value	uint32	-	≥ 0
	Maximum CW value	uint32	-	≥ 0
	Idle time	uint32	μs	≥ 0
LBT statistics	Channel clear ratio	float	-	0-100 %
	Channel clear avg. energy	float	dBW	float range
	Channel occupied ratio	float	-	0-100 %
	Channel occupied avg. energy	float	dBW	float range

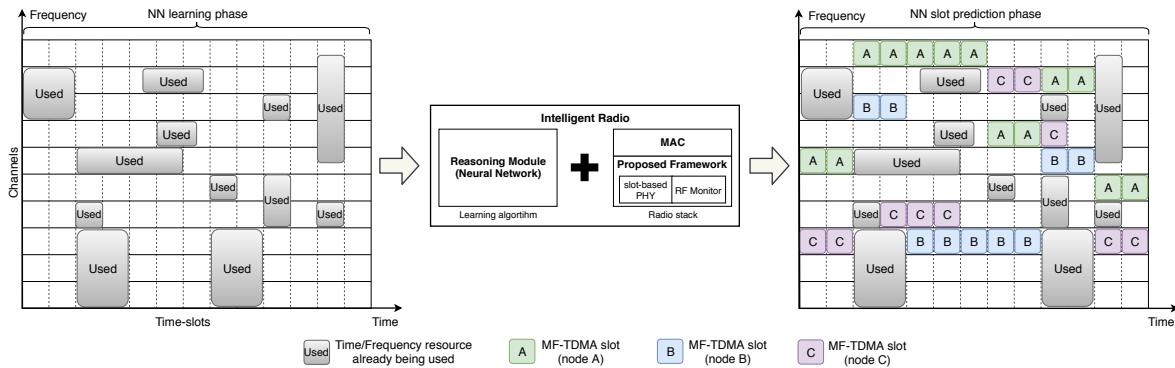



Fig. 13.   Use case scenario: MF-TDMA free slot prediction.

communications, which often results in cross-technology interference, i.e., interference from spatially close concurrent transmissions that overlap in time and frequency, an event known as collision.

In order to mitigate cross-technology interference, in this use case, the proposed framework is used to implement an intelligent Multi-Frequency Time-Division Multiple Access (MF-TDMA) like network in a congested unlicensed spectrum. Here, the slot-based PHY and the RF Monitor modules are integrated with a MAC layer and a reasoning module featuring an Neural Network (NN) model.

The NN is trained to improve slot scheduling (i.e., time/frequency allocations) by predicting transmission patterns, and consequently avoiding traffic from other nodes regardless of the technologies they are using. By observing the spectrum, through statistics received from both the RF Monitor and PHY modules, the NN is able to do online learning and predict the behavior of other nodes, finding free slots some time in advance. Based on the slot usage predictions provided by the reasoning module, the MAC layer can select in advance the best channel and the necessary COT and MCS to deliver the traffic load while not causing interference (i.e., collisions) to other radios. Figure 13 shows how the integration of the

proposed framework with a reasoning module can be used to implement an intelligent spectrum sharing scheme.

It is important to notice that any type of NN model (e.g., Deep Convolutional Networks, autoencoders, Support Vector machines, etc.) could be employed in this use case, as the framework is totally independent of the approaches (i.e., software and/or hardware) used in the upper layer's implementation.

B. Intelligent LTE-LAA and Wi-Fi Coexistence

LTE-LAA enables the deployment of LTE networks in the unlicensed 5 GHz spectrum band. However, when coexisting with LTE-LAA, the performance of Wi-Fi networks primarily relies on how the LTE-LAA parameters are configured, especially parameters such as channel, COT, idle time and CW size that govern the transmission opportunities in LTE-LAA networks.

In order to provide fairer coexistence (i.e., provide fair opportunities for all technologies to access the medium), LTE-LAA parameters must be dynamically configured based on specific statistics like collision counters, different channel utilizations, i.e., the traffic load present in a channel, etc. In this use case, the proposed framework is used to implement an intelligent LTE-LAA base station (BS). Here, the slot-based PHY and the LBT modules are integrated with a Reinforcement Learning (RL) module, which can feature algorithms like Multi-Armed Bandit or Q-Learning, to maximize the overall capacity performance through an efficient coexistence.

The RL module learns how to estimate the activity of Wi-Fi users and consequently adjust the LTE-LAA parameters for several traffic conditions. The main goal is to determine a policy by which the LTE-LAA BS can intelligently choose the optimum channel, COT, idle time and CW size based on collision statistics and measurements taken during the sensing period. Figure 14 shows how the integration of the proposed framework with an RL module can be used achieve efficient RAT coexistence. In the figure, we see in the upper part the sensing period, where the RL module learns about the medium usage, i.e., how often and how long others networks access the channels. Based on what the module learned, we see in the lower part of the figure that the LTE-LAA node is able to

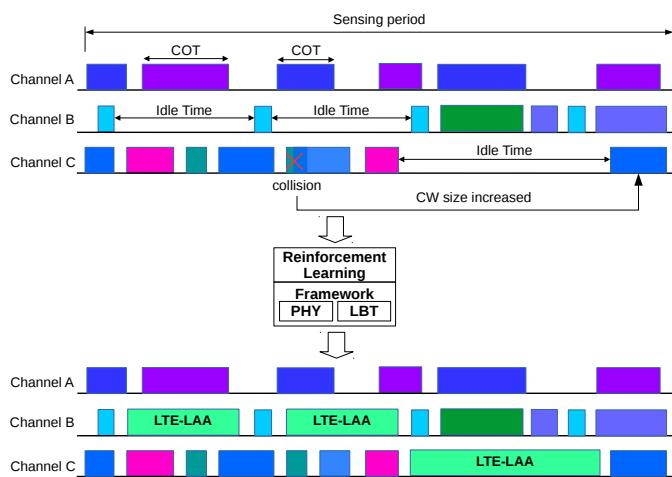



Fig. 14.   Use case scenario: Intelligent RAT Coexistence.

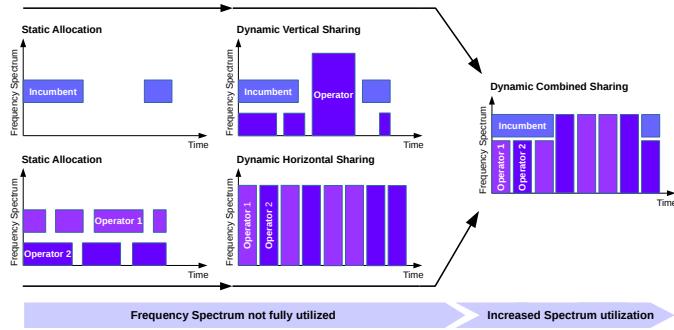



Fig. 15.   Use case scenario: Coordinated Spectrum Sharing Schemes.

select channels that can accommodate its traffic load needs.

C. Coordinated Spectrum Sharing Schemes

The adoption of spectrum sharing mechanisms is expected to bring critical benefits to the next generation of wireless networks. For example, spectrum sharing can unlock additional, currently underutilized, spectrum bandwidths, make it possible to introduce new deployment scenarios and improve spectrum utilization [73, 74].

There exists two main approaches for spectrum sharing, namely, uncoordinated and coordinated. Uncoordinated approaches make use of contention-based mechanisms like the LBT in LTE-LAA and Carrier-sense multiple access with collision avoidance (CSMA/CA) in Wi-Fi. However, at high traffic loads, as users (i.e., radios) are always competing for access to the medium without any cooperative access planning and organization, the number of collisions can be very high, which increases the radios' backoff period and as consequence, drastically reduces their throughput. Therefore, such approach is far from optimal in terms of spectrum efficiency and QoS guarantees. The low spectrum efficiency and the lack of QoS guarantees are its main disadvantages.

On the other hand, coordinated spectrum sharing makes use of a centralized database or over-the-air protocols to define a cooperative access planning and organization among users, i.e., radios. The efficiency and robustness of spectrum sharing users can be hugely increased by exploring tight coordination and time synchronization among them. Coordinated sharing has the opportunity to introduce new sharing paradigms such as vertical, horizontal and dynamic (combined) spectrum sharing, as depicted in Figure 15. By analyzing Figure 15, it is clear that a better spectrum utilization can be achieved by a dynamic spectrum sharing coordination among users. The tight coordination of time and frequency resources among users can deliver QoS levels similar to that of users with exclusive allocations of spectrum bandwidth. Through coordinated spectrum sharing, primary users (e.g., incumbents) can get access to guaranteed resources (i.e., access to the medium), while any spectrum not being used by them can be dynamically allocated among secondary users.

In this use case, the slot-based PHY with its real-time configurable parameters (e.g., Tx PHY BW, transmission times-

tamp, COT) can be used to devise and assess the performance of new and disruptive sharing protocols and models, deployment scenarios, etc. For example, Game theoretic models can be used to solve spectrum management problems like spectrum trading, spectrum sharing, interference avoidance, power allocation, etc. [75, 76]. In this use case, a central unit, employing a Game theoretic model, would receive requests for time and frequency resource allocation from highly heterogeneous users with different QoS demands and based on some QoS criteria like average throughput, delay, etc. the central unit would try to maximize the system-wide QoS following some pricing mechanism for resource allocation [76], i.e., the central unit optimizes the spectrum resources in order to meet the users' QoS requirements [77].

VIII. Experiment Results

In this section we present some experimental results in order to demonstrate the effectivenesses and usability of the proposed framework. All the experiments presented here were carried out with the framework running on servers with Intel Xeon E5-2650 v4 CPUs (@2.2 GHz,30 M cache, 9.60 GT/s QPI, Turbo, HT, 12 Cores/24 Threads, 105 Watts) with 128 GB of RAM memory connected to x310 USRPs with 10 Gigabit Ethernet links, and equipped with CBX-120 RF daughterboards [78]. These RF daughterboards operate from 1200 up to 6000 MHz with a bandwidth of 120 MHz [78]. A PHY BW of 5 MHz is selected for the experiments presented in this section, which in frequency corresponds to 25 RBs * 12 subcarriers * 15 KHz (subcarrier separation in frequency domain) = 4.5 MHz of useful bandwidth, i.e., the bandwidth really occupied by the transmission of data.

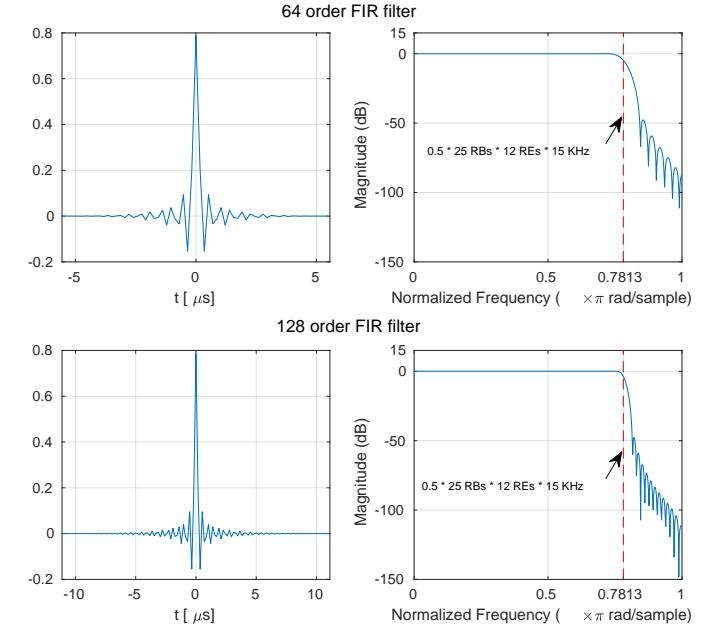



Fig. 16.   Impulse and frequency responses of the designed filter for f-OFDM with two different filter orders and bandwidth equal to 25 RBs.

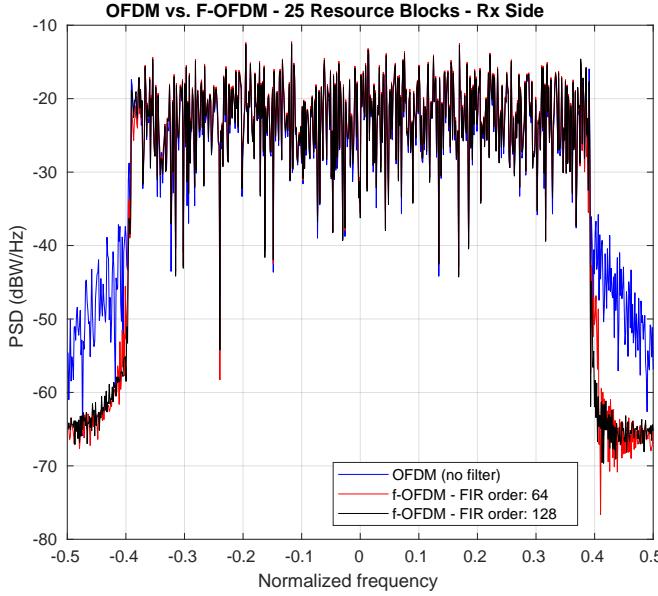



Fig. 17.   OFDM and Filtered-OFDM OOB emissions comparison at Rx side.

A. Experiment with Filtered-OFDM waveform

Next we present a few comparison results showing how OOB emissions can be mitigated with the use of a FIR filter added to the PHY Tx processing chain, just after the OFDM symbol modulation. The spectrum shaping filters employed in the following experiments are obtained by windowing the Sinc function, $p(n)$, as defined in (2) with the window function, $w(n)$, given in (3).

Figure 16 shows the base-band impulse response of the designed filter with bandwidth (i.e., NPRB) equal to 25 RBs and NFFT equal to 384. The upper and lower parts of the figure show the impulse and frequency responses of 64-th and 128-th order filters respectively. It can be noticed that the main energy of the filters is confined within the the main lobe, centered around 0 μs, and consequently, the resulting ISI stays withing tolerable levels. The figure also shows the 3 dB cutoff frequency (red-dashed lines) of the filters, which, as designed, happens around half of the useful bandwidth, i.e., 4.5 MHz / 2 = 2.25 MHz. As expected, the 128-th order filter presents a steeper transition region, which results in less interference to adjacent channels a better frequency-localization when compared with OFDM.

First, we present some results collected at the Rx side of the proposed slot-based PHY. Figure 17 depicts the power spectral density (PSD) of the f-OFDM signal and compares it with an OFDM signal. The figure shows the comparison of OOB emissions between the OFDM and Filtered-OFDM waveforms at the receiver side of the slot-based PHY for two different FIR filter orders, 64 and 128 respectively. For this result, the FIR filter, either with order 64 or 128, is added to the Tx processing flow (i.e., OFDM symbol modulation), see Figure 8. The figure is obtained by collecting IQ samples at the receiver side of the slot-based PHY, after Rx front end-processing (RF and digital processing) and after the slot (i.e., subframe) is

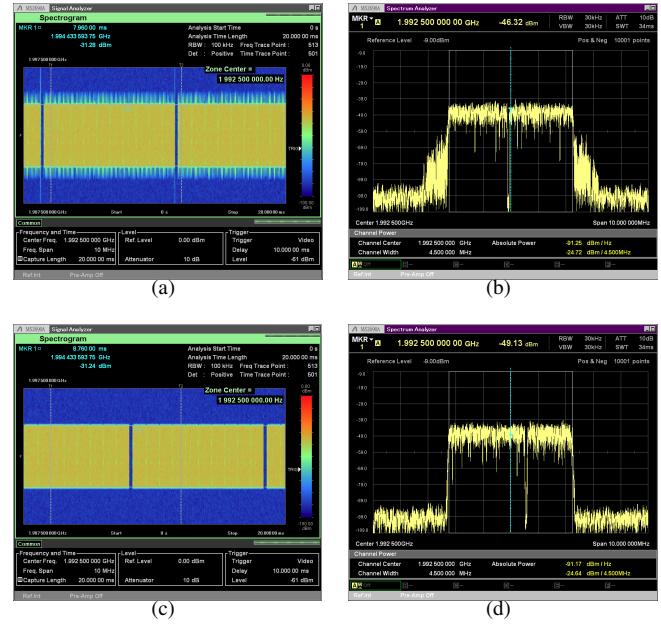



Fig. 18.   Comparison between OFDM and Filtered-OFDM. (a) Spectrogram with OFDM (i.e., no filter). (b) Spectrum with no filter enabled. (c) Spectrogram with 128 order FIR filter enabled. (d) Spectrum with 128 order FIR filter enabled.

synchronized and aligned, therefore, this is the signal fed into the demodulation module at the receiver side of the slot-based PHY. Table VI compares some approximate measurements of power spectral density (PSD) and filter processing time.

As can be seen by analyzing Figure 17 and Table VI, the filter reduces the OOB emissions from around -40 dBW/Hz to less than -52 dBW/Hz at the edge of the transmitted signal, around 0.4fs, where fs is the sampling rate. As can also be noticed in Figure 17, the 128 order FIR filter has, as expected, a sharper transition region when compared to the 64 order FIR filter. Table VI also shows the processing time for each one of the FIR filter orders. The processing time presented in the table considers the time it takes to filter a 1 ms long slot. The values are the result of the average of 10000 slot transmissions. For a PHY with 25 RBs of BW, a 1 ms long slot is composed of 5760 complex samples, i.e., 5760 I/Q samples. Each complex sample is represented by 32 bits (i.e., 4 bytes, with 2 bytes representing each one of the I and Q components), which means the filter has to process 23040 bytes.

In Figure 18 we compare the spectrogram and spectrum of OFDM and f-OFDM transmissions collected with an Anritsu

TABLE VI

OFDM VS. F-OFDM OOB EMISSION AND PROCESSING TIME

COMPARISON.

FIR order	PSD @ 0.4fs [dBW/Hz]	PSD @ 0.5fs [dBW/Hz]	Processing time [ms]
No Filter	-40.83	-56.21	-
64	-52.12	-64.52	0.12
128	-55.39	-65	0.24

TABLE VII

DIFFERENCE BETWEEN P_{ADJ} (NO FILTER) AND P_{ADJ} (64/128 ORDER FILTERS).

Offset	L1 (-5 MHz)		U1 (+5 MHz)		L2 (-10 MHz)		U2 (+10 MHz)		Tx gain
FIR order	64	128	64	128	64	128	64	128	[dB]
	15.69	15.97	14.82	14.66	0.01	0.16	0.29	0.27	3
	18.5	18.82	17.77	18	0.38	0.6	0.09	0.21	6
	20.43	20.39	19.06	19.17	0.56	0.03	0.20	0.55	9
	21.88	21.27	21.24	20.95	0.51	0.21	0.30	0.49	12
Difference	23.09	22.89	22.47	22.28	0.22	0.02	0.21	0.54	15
[dB]	23.11	22.71	22.1	21.84	0.65	0.26	0.47	0.40	18
	22.9	22.59	21.07	20.73	0.48	0.21	0.32	0.13	21
	19.38	18.94	17.52	16.94	0.48	0.14	0.25	-0.02	24
	11.21	10.8	10.51	10.19	0.67	0.36	0.6	0.35	27
	3.29	3.02	2.74	2.51	1.18	0.68	1.12	0.51	30

Fig. 20. ACLR comparison between OFDM and Filtered-OFDM with Tx gain equal to 18 dB. (a) OFDM with no filter enabled. (b) 64 order FIR filter enabled.

MS2690A Signal Analyzer. The figures were collected with a Tx center frequency of 1.9925 GHz, Tx gain of 3 dB with the USRP Tx output connected to the signal analyzer through a cable with 20 dB of attenuation. It is easily seen that the OFDM side lobes (i.e., OOB emissions) are reduced with the use of the 128 order FIR filter.

The USRP expects to be fed with samples at a rate close to *fs*, however, if the processing time of the Tx chain increases so that it can not deliver samples fast enough (i.e., at a rate close to *fs*), then the USRP starts outputting 'U's (Underflows) at the command line, meaning, that the slot-based PHY transmitter is not supplying samples fast enough. Therefore, in order to keep the Tx processing flow close to the specified sample rate, the processing time of the filter must be as low as possible. During the experiments with the current setup, we also tested higher order filters like 256 and 512, however, the processing time increased to the point where the slot-based PHY transmitter was not being able to supply samples fast enough to the USRP. For smaller PHY BWs the number of complex samples making up a slot is smaller, which makes it possible to use higher order filters. Other FIR filter orders can be easily added to the slot-based PHY by just updating the table with filter coefficients.

Another kind of measurement that was taken in order to show the improvement provided by the f-OFDM waveform is known as Adjacent Channel Leakage Ratio (ACLR). ACLR is an important characteristic of wireless transmitters and is defined as an important wireless metric by national laws regulating radio standards as well as by 3GPP and IEEE.

ACLR is defined as the ratio between the total integrated adjacent channel average power, P_{adj}, centered at upper and lower channel frequencies and the total integrated reference channel average power, P_{ref}, centered at the assigned reference channel frequency (i.e., in-band power), where the powers are measured after a receiver filter [79]. The ACLR metric is defined as

$$\mathrm{ACLR(dBc)} = 10\log_{10}\left(\frac{P_{\mathrm{adj}}}{P_{\mathrm{ref}}}\right) = P_{\mathrm{adj}}(\mathrm{dBm}) - P_{\mathrm{ref}}(\mathrm{dBm}).$$
(7)

In digital communications systems, the power that leaks

Fig. 19. ACLR measurements for different Tx gains and filter orders.

from a transmitted signal into adjacent channels can interfere with transmissions in the neighboring channels and decrease the neighbor system's performance. For LTE, the ACLR metric verifies that transmitters are performing within specified limits defined in the 3GPP specifications [80, 81].

Figure 19 presents the adjacent channel average power, P_{adj} in dBm, for different frequency offsets, Tx gain values and FIR filter orders. The figure also shows the reference channel average power P_{ref} in dBm. The measurements were also taken with the Anritsu MS2690A Signal Analyzer for a setup where Tx center frequency is set to 1.9925 GHz, COT is set to 100 ms, which means 100 1 ms-long slots are transmitted in a row, with an idle time of 1 ms between consecutive transmissions and MCS is set to 0. The Tx gain is changed from 3 up to 30 dB in steps of 3 dB for three different cases: no filter, 64 and 128 order FIR filters. The connection between the USRP and the Spectrum Analyzer is done through cable and attenuation of 20 dB is applied to the signal.

Table VII shows the difference between P_{adj} when no filter is used and P_{adj} when 64 or 128 order FIR filters are used. As can be seen, the power leakage on adjacent channels with frequency offset of +/- 5 MHz is greatly decreased when using any of the two filter orders. For these cases, the leakage reduction can be as high as 23.11 dB for a Tx gain of 18 dB (probably the most linear region of the Tx power amplifier). It can also be noticed that the leakage reduction for both filters orders is quite close. This means that the 64 order filter can be employed without any huge impact on the leakage onto adjacent channels while reducing the Tx flow processing time. On the other hand, the power leakage on adjacent channels with frequency offset of +/- 10 MHz is marginally decreased (at most 1.18 dB reduction), meaning that the leakage level on those channels is already very low even without filter. Figure 20 depicts one example of the ACLR measurement for a Tx gain of 18 dB with OFDM filtering disabled and enabled (64 order FIR filter) respectively. It is easily noticeable that the OFDM skirt is mitigated when the filter is enabled.

All the experiment results presented in this subsection show that the designed filters are able to attenuate out-of-band emissions without affecting in-channel performance.

B. Experiments with closer coexistence

In this section we present the results of experiments where we had a closer coexistence among nodes, i.e., the frequency offset between adjacent channels was made smaller. As known, the OFDM waveform is not well localized in frequency and therefore, it needs guard bands (GB) at both ends of the transmitted bandwidth to protect adjacent channels from spillage (also known as spectrum spread or leakage) of the OFDM signal into their bands, i.e., the GBs are used so that the interference being caused to adjacent channels is reduced. For LTE systems, these GBs account for 10 % of the available bandwidth and their purpose is to meet OOB emission level requirements. For example, for a PHY with bandwidth of 10 MHz, GBs of 750 KHz are used, which accounts for 1.5 MHz in total of wasted spectrum. Therefore, the purpose of the

experiments described in this section is to show that closer coexistence and consequently improved spectrum utilization, i.e., less spectrum wastage, is possible with well designed filters.

The setups used in the experiments are depicted in Figure 21. In Figure 21 (a), we show the setup used to asses the closer coexistence. In this setup, as can be seen in the figure, three nodes simultaneously transmit data to each other on three different adjacent channels. Differently from standard LTE, where there would be a 10 % GB, the adjacent channels are now only 4.5 MHz apart from each other, which means that there is no GB at all between adjacent channels. The channel spacing of 4.5 MHz comes from the fact that the useful bandwidth of a 5 MHz PHY is equal to 25 PRBs $\times$ 12 subcarriers per PRB $\times$ 15 KHz of subcarrier spacing. The closer coexistence experiment is split into three sub-experiments: (i) no filtering enabled in any of the three nodes, (ii) 64 order filters enabled in all the three nodes, and (iii) 128 order filters enabled in all the three nodes. For quantitative comparison, we used the simplex case, where one node transmits to the other on one given channel and there is no other transmissions on adjacent channels. This setup is shown in Figure 21 (b).

For all the experiments, Tx and Rx gains were set to 20 dB and COT of 10 ms, i.e., 10 slots of 1 ms with a gap of 0.5 ms between subsequent transmissions was used. We use 20 dB for Tx and Rx gains so that the received signal quality (i.e., SINR) would be good enough for the slot-based PHY to reach high MCS values. The gap of 0.5 ms is the lowest possible gap value with the current implementation of the slot-based PHY due to the latencies introduced by the 0MQ bus and the communication with the USRP. This gap is selected so that the channel occupancy time is higher than 95 % and with that, the interference to the DUT will also happen more than 95 % of the time. We used a testbed with constant 60 dB path loss for all node pairs is used, meaning that the RF signals generated by one node experience 60 dB attenuation before reaching anyone of the other nodes. The testbed has a bandwidth of 20 MHz centered at a frequency of 1 GHz, which means signals with frequencies less than 0.99 GHz and/or greater than 1.01 GHz are heavily attenuated.

We quantitatively evaluate the closer coexistence and the impact of the filtering on the closer coexistence setup with two metrics, namely, packet reception rate (PRR) and average CQI, which are reported by the receiver. The PRR is calculated by counting the total number of detected slots (i.e., slots that had the PSS correlation peak higher than the predefined detection threshold) and the number of errors (i.e., the total number of synchronization and decoding errors). The results were collected during the transmission of 10000 frames with 10 1 ms-long slots for all MCS values (0 to 28).

Figures 22 and 23 present the PRR and average CQI results respectively. By analyzing figure 22, it becomes clear from observing the results that the interference is only noticeable for MCS values greater than 22, as the PRR is always equal to 100 % for MCS values ranging from 0 to 22. Therefore, the

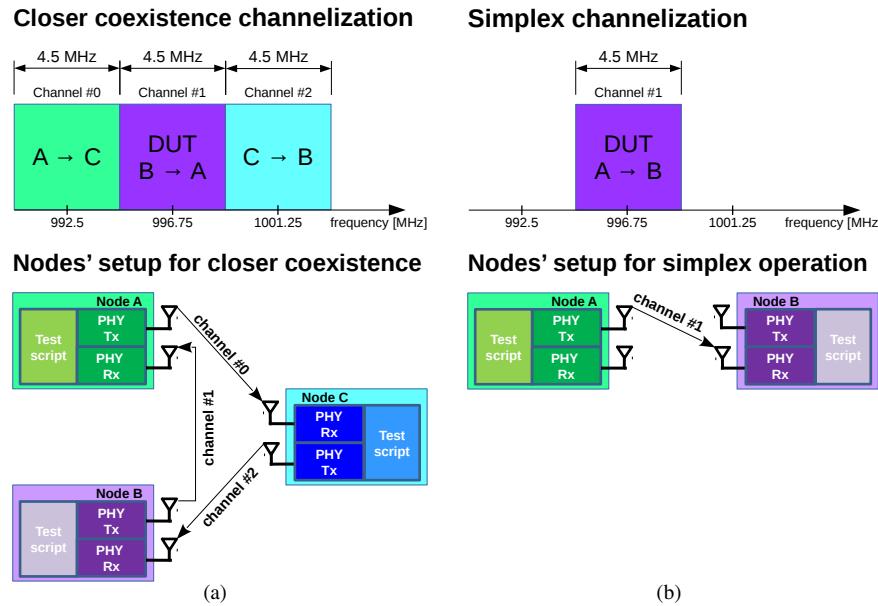



Fig. 21. Closer coexistence experiments. (a) Channelization and nodes' setup for the closer coexistence experiment. (b) Channelization and nodes' setup for the simplex operation experiment.

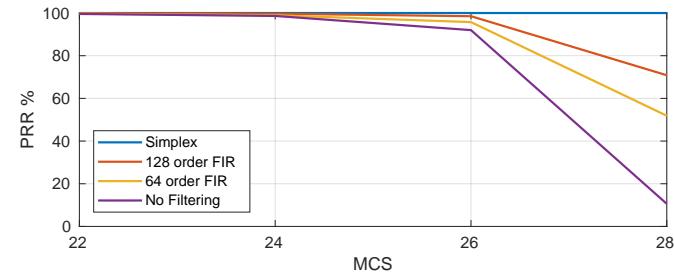
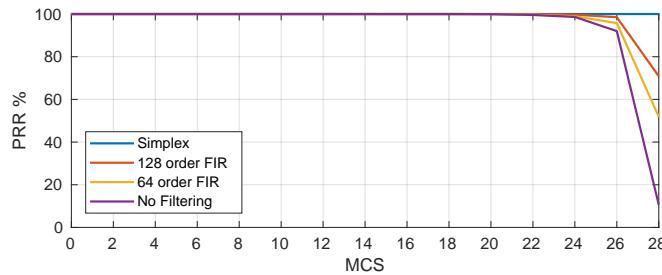



Fig. 22. Packet Reception Rate for the experiment with no spacing between adjacent channels.

filtering will be only effective, and its provided improvement noticeable, for MCS values greater than 22. This is due to the fact that the FEC redundancy (i.e, Turbo coding) added to the transmitted data is high enough to make it much less susceptible to the interference caused by the leakage coming from the adjacent channels. As expected, in the closer coexistence sub-experiment with no filter enabled, the device under test (DUT) suffers the highest interference as it has two radios simultaneously transmitting in the upper and lower adjacent channels. Additionally, when 64 order FIR filters are enabled in all nodes, the PRR increases from 10 % (case with

no filtering) to approximately 52 % and for the case where the 128 order FIR filters are enabled in all the nodes, the PPR increases to more than 70 %.

In figure 23 average CQI results are presented. The top plot shows the result for all the cases while the bottom plot excludes the average CQI for the Simplex case so that the results for the cases when filtering is enabled can be analyzed clearer. As can be seen, the average CQI increases from 8.1 to more than 9.3 when the 128 oder FIR filter is employed, meaning that the interference being caused by the spillage decreases, consequently increasing the SINR, which in turn

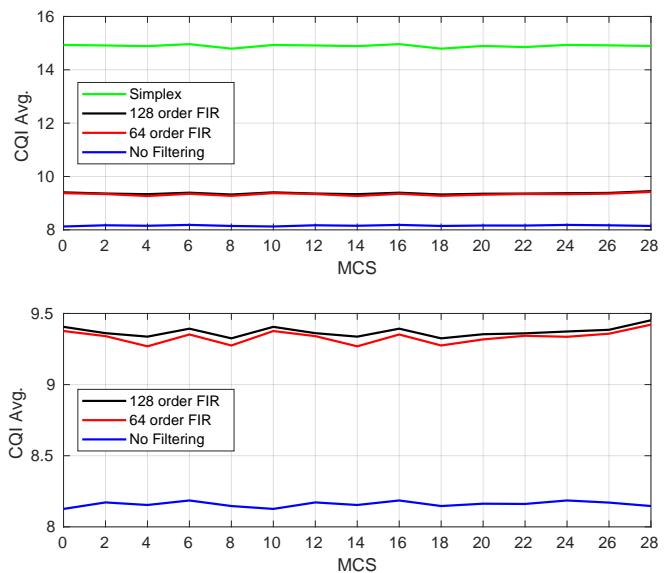



Fig. 23. Averaged CQI for the experiment with no spacing between adjacent channels.

increases the CQI. CQI is a quantized version of the SINR and indicates the highest MCS at which the block error rate (BLER) for the current channel does not exceed 10 % [82]. Moreover, it can be noticed that the average CQI for the 128 order FIR case is slightly higher than the one for the 64 order FIR filter. It is also observable that even with the filtering enabled, the resulting CQI is still lesser than that observed for the simplex case. It is explained by two facts: (i) the FIR filter orders employed here can not eliminate the OOB completely and some of it still spills over the adjacent channels. Besides that, the filter is added before the power amplifier (PA), which amplifies the reminiscent leakage. (ii) the filter's frequency response is not totally flat throughout the passband, especially at the edges, where it attenuates the upper and left parts of the transmitted signal. These two facts directly impact on the SINR, which in turn, impacts the CQI.

With these experiments, we show that when GBs are removed, the filtered-OFDM waveform achieves MCS 28 with PRR above 70 %, which is 60 % better than the case with no filter enabled. Therefore, with the right filtering and just enough channel spacing, the spectrum utilization can be improved.

C. Experiment with the LBT module

In this section we describe the results of an experiment with the LBT module. The main goal of this experiment is to show that LBT can decrease the number of collisions and consequently improve the coexistence among radios operating at the same frequency.

The experiment consists of three different steps. For all the three different steps we want to assess the coexistence performance in terms of CQI and RSSI measurements. The CQI is a quantized version of the signal-to-interference-plus-noise ratio (SINR) measurement that indicates the channel quality. The receiver reports CQI measurements back to the transmitter indicating the data rate supported by the channel at that instant. This measurement report helps the transmitter to choose the optimum modulation and code rate, i.e., MCS, for its transmission. CQI is quantized as shown in [82]. The SINR is calculated as follows

$$\text{SINR} = 10\log_{10}\left(\frac{\text{RSRP}}{\text{noise\_power}_{\text{AVG}}}\right), \qquad (8)$$

where RSRP means Reference Signal Received Power and is the linear average of reference signal power across the specified bandwidth (in number of PRBs). When a collision happens, the average noise power increases as the interference being caused by other radios' transmissions is perceived by the receiver as effective noise. Consequently, the SINR decreases, which in turn decreases the CQI. The RSRP is calculated as

$$\text{RSRP} = \frac{\sum_{k=0}^{P-1}|Y_p(k)|^2}{P}, \qquad (9)$$

where $Y_p(k)$ represents the received pilot symbol values in the frequency domain, i.e., the value of resource elements (RE) carrying Cell-Specific Reference Signal (CRS) over the entire

PHY bandwidth, and $P = 8 \times$ number of PRBs used by PHY. The average noise power over all REs carrying pilot symbols is estimated as [83, 84]

$$\text{noise\_power}_{\text{AVG}} = \frac{1}{P}\sum_{k=0}^{P-1}|Y_p(k) - \tilde{H}_p(k)X_p(k)|^2, \quad (10)$$

where $X_p(k)$ represents the known transmitted pilot symbol values and $\tilde{H}_p(k)$ is the estimated channel response for the RE occupied by the k-th pilot symbol. The channel response at REs carrying pilot symbols is obtained by dividing the received pilot symbols by their expected values,

$$\tilde{H}_p(k) = \frac{Y_p(k)}{X_p(k)} = H_p(k) + \text{noise}. \qquad (11)$$

The other performance metric used in the experiment is the RSSI. It measures the average total received power observed over the whole slot (i.e., 1 ms) duration. The RSSI is calculated as

$$\text{RSSI} = 10\log_{10}\left(\frac{\sum_{n=0}^{N-1}|y(n)|^2}{N}\right), \qquad (12)$$

where $y(n)$ is the received slot signal in time domain, N is the slot size in number of complex samples, which is, for example, equal to 5760 samples for a 5 MHz PHY. When a collision happens, the transmitted signals (i.e., slots) overlap and consequently, the measured RSSI will be higher as the received signal is the combination of several transmissions that happen at the same time and frequency.

For all the three steps of this experiment we used a testbed that provides a constant 60 dB path loss for all radio pairs and a bandwidth of 6 MHz centered at a frequency of 1 GHz, i.e., transmitted signals with bandwidth wider than 6 MHz are heavily attenuated. The radios' Tx and Rx gains are set to 20 dB, COT is set to 25 ms, i.e., 25 slots are transmitted in sequence without any gap between subsequent slots, and the idle time is set to 10 ms, i.e., the interval between COTs. We transmitted more than 180×10^3 slots with MCS equal to 0 for each one of the steps of the experiment.

For the first step, we assess the performance when there is no collision involved. In this step there are two radios working in simplex mode, i.e., one radio transmits and the other one only receives. In this step there are no collisions. As can be seen in Figure 24 (a), RSSI is constant at around 25 dBW and CQI varies between 11 and 15 throughout the trial.

In the second step, we want to assess the performance when two radios compete for access to the medium (i.e., two radios operating in full-duplex mode at the same central frequency) without any contention mechanism. Figure 24 (b) shows that

TABLE VIII

LBT EXPERIMENT MEASUREMENTS.

Step	CQI		RSSI		Collision %
	Mean	Variance	Mean	Variance	
Simplex	14.77	0.37	-25.33	0.0003	0
LBT Disabled	12.79	11.02	-25.10	0.2258	31.813
LBT Enabled	13.78	1.27	-25.24	0.0091	0.038

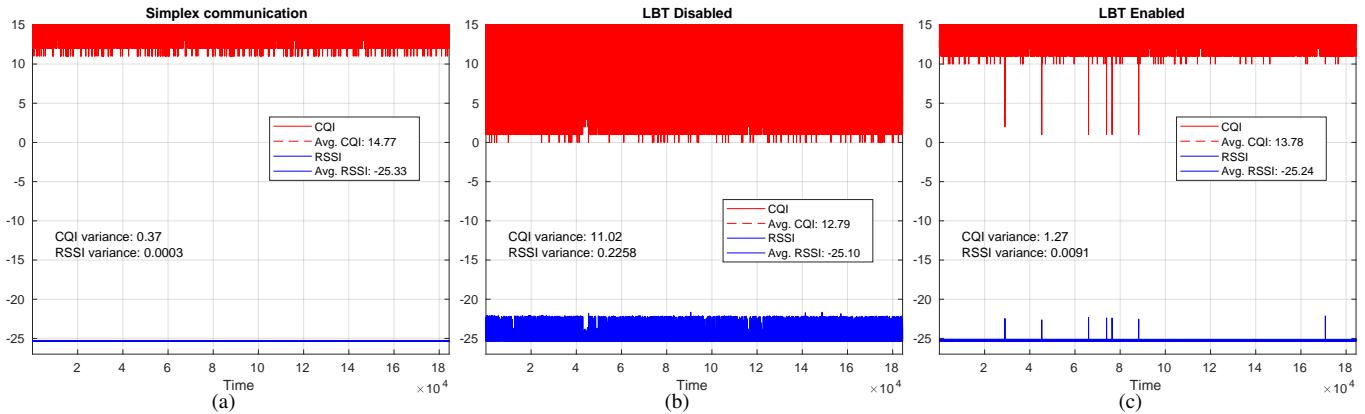



Fig. 24.   Comparison when LBT is disabled and enabled. (a) Simplex operation: one radio transmits and the other one only receives. (b) Two radios transmitting on the same channel without any contention mechanism. (c) Two radios transmitting on the same channel with LBT enabled.

RSSI is not constant anymore, varying between -25 and -21 dBW while CQI varies between 0 and 15 throughout the trial. The fluctuation in both CQI and RSSI values are due to collisions. As a way to estimate the number of collisions during the step, we count as a collision CQI measurements with value less than 11, which is the minimum CQI value for the simplex communication case, where no collisions happen. By using this approach, the percentage of collisions is equal to 31.813 %.

The third step is similar to the second step with the only difference being the use of a contention mechanism, i.e., LBT. In this step we assess the performance when LBT is enabled with a threshold of -70 dBW, CCA of 173.26 μs and a random backoff of at most 32 CCA intervals. Figure 24 (c) shows that RSSI is much more constant, staying around -25 dBW with very few points where it goes to -21 dBW. The CQI behavior is also closer to that depicted in (a), with CQI varying between 10 and 15 and very few points where it drops to 0 throughout the trial. The percentage of collisions is equal to 0.038 % using the same approach described earlier to estimate the number of collisions. Table VIII summarizes the results of the three steps of the experiment.

Based on the results of this experiment, it is clearly seen that either CQI or RSSI measurements (or even both of them) could be used as an approach to detect collisions. For example, these measurements could be employed in LTE systems to detect collisions more effectively in a timely manner and consequently, decrease the inherent latency of those systems. In LTE systems, due to the inherent latencies introduced by the LTE protocol stack, the HARQ feedback associated to a certain subframe is received at least 4 ms after its transmission time [85].

D. Experiment with MF-TDMA feature

In this section we show the capability of the slot-based PHY to work in MF-TDMA mode with the use of the transmission timestamp field in the Tx control message. MF-TDMA is a two-dimensional multiple access scheme that combines frequency division with time division access to

the medium. Due to its efficiency and flexibility, MF-TDMA scheme has been widely used in several communications systems such as very small aperture terminal (VSAT) satellite communications systems [86]. A MF-TDMA communications system is composed of several channels, where each one of the channels is divided into a number of time slots. Such scheme ensures adaptability and improves the radio resources utilization for multi-user and multi-service applications [86].

Figure 25 shows two spectrograms collected during 40 ms over a 31.25 MHz bandwidth (the Anritsu MS2690A Signal Analyzer does not allow a bandwidth of 30 MHz, being 31.25 MHz the closest configurable value) with the slot-based PHY set to operate with six 5 MHz channels (i.e., 25 RBs) with a channel spacing of 2.5 MHz. In this experiment, one radio transmits a random number of slots at randomly selected channels. The channel number and number of transmitted slots, i.e., COT, are randomly selected between the ranges 0-5 and 1-3 respectively.

Figure 25 (a) presents the spectrogram for the case with no filtering enabled (i.e., OFDM waveform) and with a gap of 0.5 ms between consecutive transmissions. As can be noticed, with the current channel spacing, the OFDM OOB emissions might cause interference to adjacent channels and consequently decrease the system's throughput. The OOB emissions can be mitigated with the use of filtering as shown next.

Figure 25 (b) depicts the spectrogram for the case when the filtered-OFDM waveform is enabled with a gap of 0.5 ms between consecutive transmissions. As can be seen, when the filtered-OFDM is used, the OOB emissions are mitigated and consequently, interference to and from adjacent channels is also mitigated. Additionally, another consequence of the filtering use is that the channel spacing could be made smaller and consequently improve the radio resource utilization performance, i.e., decrease the spectrum wastage.

Figure 26 depicts throughput measurements taken with the slot-based PHY for 1.4, 5 and 10 MHz PHY BWs, several MCS and Duty Cycle (DC) values working in full-duplex mode (i.e., the slot-based PHY is simultaneously transmitting

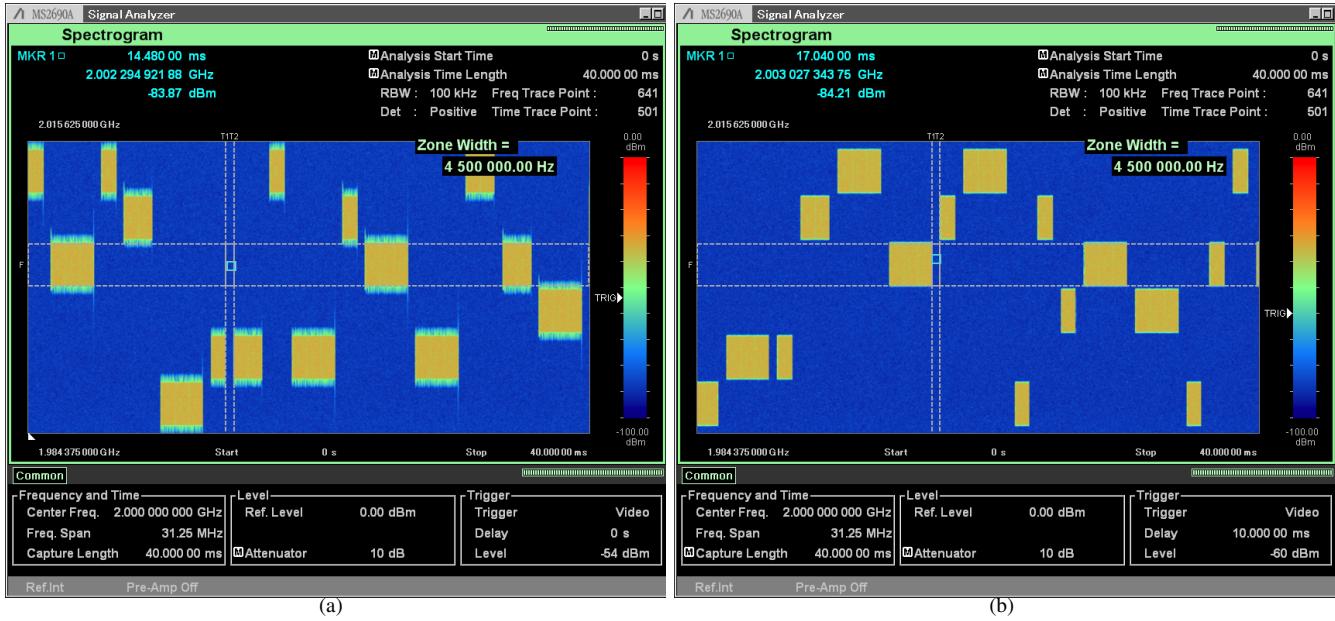



Fig. 25.   Comparison of MF-TDMA feature without and with filtering. (a) OFDM without filtering. (b) 128 order FIR filter enabled.

and receiving). We use full-duplex mode in order to check if it impacts somehow the measured throughout once in full-duplex mode the slot-based PHY is being fully utilized. The measurements were taken for COTs of 1, 10 and 100 ms with a gap of 0.5 ms between subsequent transmissions. This means that the DC increases as the number of slots sent in a row also increases. The throughput was averaged over 10 measurement intervals of 10 seconds each. During one measurement interval (i.e., 10 s) the number of received bits is counted and then divided by the interval to produce the throughput measured during that interval. For this experiment, we have also enabled 128-th order FIR filtering. In the figure, for comparison reasons, we also add the theoretical maximum throughput achieved by the *Streaming* mode, where there is no gap between subsequent transmissions, i.e., a DC equal to 100 %. The theoretical maximum throughput is calculated diving the TB size in bits for each MCS by 1 ms. As expected, the measured slot-based PHY's throughput approaches the theoretical maximum throughput for all MCS and PHY BW values as the DC increases, yielding more than 36.5 Mbps for a PHY BW of 10 MHz and MCS 28. Additionally, the operation in full-duplex mode has no visible impact on the achieved throughput. This is due to the powerful server, with 12 cores, used to run the slot-based PHY.

E. slot-based PHY Profiling

   In this section we present some measurements regarding the average Tx and Rx processing times of the proposed slot-based PHY. Table IX shows the average processing times for PHY BWs of 1.4, 5 and 10 MHz for MCS values of 0, 10, 18, 27 and 28 which use QPSK, 16QAM and 64QAM modulations as indicated in the table. With a PHY BW of 1.4 MHz it is not possible to reach the highest MCS value, i.e., 28, due to the

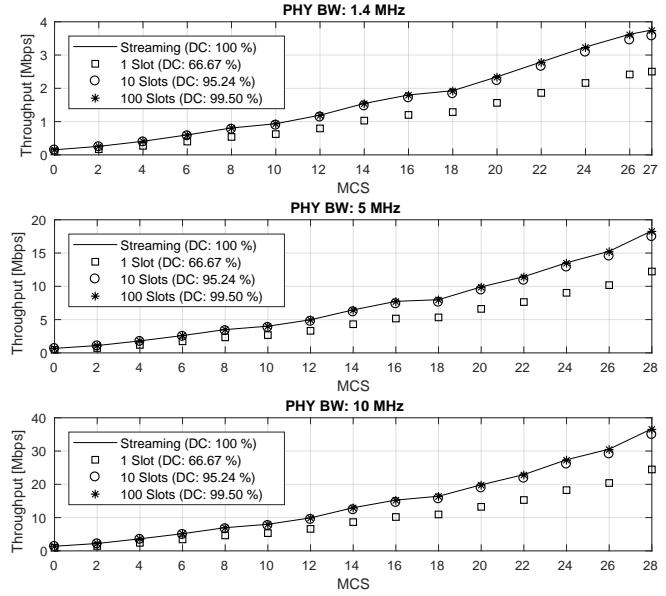



Fig. 26.   Slot-based PHY throughput for 1.4, 5 and 10 MHz PHY bandwidths, different MCS/DC values and 128-th order FIR filtering.

fact that the slot structure uses two OFDM symbols to carry control information and another two OFDM symbols to carry PSS and SSS signals (see Figure 5), which makes the effective payload size (i.e., the number of data bits an 1 ms long slot can transport) smaller than the transport block size.

   The processing times were averaged over 10000 transmissions with COT set to 1 ms, i.e., only 1 slot for each new transmission and a gap of 0.5 ms between subsequent transmissions. For these measurements we set both Tx and Rx gains to 20 dB so that the received signal quality (i.e., SINR)

would be good enough for the PHY to reach high MCS values.

The Modulation, *Mod.*, column lists the average time it takes to process control/data bits (i.e., channel coding, rate matching, scrambling, modulation, precoding and resource grid mapping) and generate a 1 ms long slot. The Filter column shows the average filtering time. As can be seen, for a given PHY BW, the filtering time is approximately constant. It shows that the filtering time does not depend on the MCS being used as, for a given PHY BW, a slot always has the same size irrespective of the MCS. The Synchronization, *Synch.*, column shows the average time it takes to correlate the received signal with the local PSS sequence, decode the SSS signal, estimate CFO, align the subframe to the buffer start (read from the USRP missing samples necessary to have a full slot in the buffer), and correct CFO. As seen from the table, the synchronization time is approximately constant for a given PHY BW, showing, as expected, that it does not depend on the MCS value. The synchronization time depends only on slot size, i.e., number of I/Q samples making up a slot. The Demodulation, *Demod.*, column lists the average time it takes for the already synchronized slot to be demodulated. As expected, the higher the MCS the higher the decoding time. That is due to the fact that the the information (data) carried by one slot is higher for high MCS values, and therefore, the processing time is longer. Another point that is also worth mentioning is that the demodulation process is the most time-consuming task carried out by PHY.

Next, we use the *psutil* library to assess CPU and memory utilization of the slot-based PHY. *psutil* is a Python cross-platform library for retrieving information on running pro-

cesses and system utilization, such as CPU, memory, disks, network, sensors, etc. [87]. Our goal with this experiment is simply to evaluate the computational expense of the slot-based PHY operation in full-duplex mode, i.e., simultaneously transmitting and receiving data. Therefore, we use a simple script connected to PHY that only transmits and receives data without any implementation of upper layer protocols. The CPU percentage scale used by the *psutil* lib is 100 % when a core is fully utilized or when one core is at 20 % load and a second one is at 80 % load. Therefore, with multicore CPUs the CPU utilization percentage can exceed 100 %. For instance, the *psutil* lib can return a CPU utilization from 0 % (idle application) up to 800 % (application fully using the cores) for CPU with 8 cores.

In this experiment 2 nodes transmit 10×10^3 1 ms-long slots with a gap of 0.5 ms between subsequent transmissions to each other on two different channels (i.e., non-overlapping frequencies) and with Tx and Rx gains of both nodes set to 20 dB. We used a testbed that provides constant 60 dB attenuation between the radios and a bandwidth of 6 MHz centered at a frequency of 1 GHz.

Figure 27 shows the CPU and memory usage of the slot-based PHY, operating at 1.4, 5 and 10 MHz bandwidths over the whole set MCS values. The results presented in the figure were obtained by averaging CPU and memory usage values sampled every 200 ms during the whole duration of the experiment. We sample CPU and memory utilization of only one of the nodes.

As expected, as the PHY BW increases, both CPU and memory utilization increase, however, there is no CPU or memory starvation issues with any of the PHY BW and MCS values. The CPU usage increases as the MCS increases. This is due to the fact that for higher MCS values (i.e., higher

TABLE IX

AVERAGE TX AND RX PROCESSING TIMES FOR DIFFERENT PHY BWS AND MCS VALUES WITH 64 ORDER FIR FILTER.

PHY BW: 1.4 MHz						
slot size: 1920 I/Q samples						
MCS	Tx processing time [ms]			Rx processing time [ms]		
	Mod.	Filter	Total	Synch.	Demod.	Total
0 (QPSK)	0.030	0.049	0.079	0.051	0.114	0.165
10 (16QAM)	0.052	0.047	0.099	0.052	0.124	0.176
18 (64QAM)	0.053	0.048	0.101	0.049	0.140	0.189
27 (64QAM)	0.065	0.047	0.112	0.046	0.196	0.242

PHY BW: 5 MHz						
slot size: 5760 I/Q samples						
MCS	Tx processing time [ms]			Rx processing time [ms]		
	Mod.	Filter	Total	Synch.	Demod.	Total
0 (QPSK)	0.072	0.120	0.192	0.185	0.166	0.351
10 (16QAM)	0.101	0.122	0.223	0.185	0.212	0.397
18 (64QAM)	0.135	0.120	0.255	0.187	0.284	0.471
28 (64QAM)	0.206	0.121	0.327	0.186	0.342	0.528

PHY BW: 10 MHz						
slot size: 11520 I/Q samples						
MCS	Tx processing time [ms]			Rx processing time [ms]		
	Mod.	Filter	Total	Synch.	Demod.	Total
0 (QPSK)	0.134	0.285	0.419	0.383	0.283	0.666
10 (16QAM)	0.175	0.283	0.458	0.384	0.407	0.791
18 (64QAM)	0.256	0.287	0.543	0.382	0.543	0.925
28 (64QAM)	0.394	0.286	0.680	0.385	0.655	1.040

Fig. 27.   CPU and memory utilization of the slot-based PHY.

Fig. 28. CPU profiling of individual components of the slot-based PHY when using a 5 MHz PHY bandwidth.

data rates), both Quadratic Permutation Polynomial (QPP) interleaver (which is part of the turbo encoding process at Tx side), synchronization and turbo decoding (both at Rx side) tasks gain weight and consume quite a lot of CPU for data processing. The CPU time spent by individual PHY functions is presented next and those results are in line with the results in [29, 88]. For a PHY BW of 10 MHz the CPU utilization is of around 75 % meaning that the processing power of only one core is being used, however, it is not being fully utilized, and most probably, leaving the other cores in idle state for large periods of time.

In the case of memory usage, it can be observed by analyzing the figure that for a given PHY BW it is practically constant for all MCS values and therefore, independent of the configured MCS value. This is the expected result as all the memory used is pre-allocated during the initialization of the slot-based PHY, i.e., there is no memory allocation during PHY operation. Based on these results, we conclude that the slot-based PHY does not exhaust CPU or memory resources as PHY BW or MCS increases.

In the following, we evaluate the CPU consumption of independent processing tasks (i.e., functions) within the slot-based PHY software. For this evaluation we use *valgrind* with its *callgrind* tool. *Callgrind* is a profiling tool that records

the call history among functions in a program's run as a call-graph through the use of runtime instrumentation [89]. By default, the collected data consists of the number of instructions executed, their relationship to source lines, the caller/callee relationship between functions, and the numbers of such calls [89].

Valgrind adds a considerable amount of instrumentation code to the application it is profiling, which makes the application 4 to 50 times slower [90]. Therefore, when running the slot-based PHY through *valgrind* with the *calgrind* tool, we noticed that when transmitting slots, PHY was not being able to feed samples at the USRP's expected sample rate and in consequence, large parts of a slot were not being transmitted due to underflows. For that reason, in order to be able to profile PHY, we decided to split the profiling into two parts, one where only the Tx thread is spawn (i.e., all Rx threads are disabled) and a second part where only the Rx threads are spawn (i.e., the Tx thread is disabled).

The setup for this experiment is the same as the one used earlier for CPU and memory profiling. In the first part, where only the Tx thread is started, the node exclusively transmits slots while PHY is profiled. For the first part, as we are only interested in the profiling of Tx related functions, there is no need for a second node receiving the data. In the second part, where only the Rx threads are started, the node with PHY being profiled receives data from a second node, which is not being profiled.

Our findings are summarized in Figure 28 for a PHY with bandwidth equal to 5 MHz and different MCS values. The figure depicts the most representative CPU consumers, i.e., functions that use more CPU processing for the two considered parts of the experiment. The left column shows CPU time spent by functions related to Tx processing, while the right column presents CPU time spent by functions related to Rx processing.

As can be seen, for Tx related functions, bit interleaving (i.e., QPP interleaver) gains weight as MCS increases, consuming approximately 65 % of CPU time for MCS equal to 28 compared to only 9 % when MCS is equal to 0. For Rx related functions, it is worth highlighting how the synchronization task consumes a great deal of CPU time for all the MCS values considered in this experiment ranging from 28 % to more than 42 % of CPU time. It is also worth mentioning how the turbo decoding task gains weight as MCS increases, going from 5 % to more than 36 % for MCS values equal to 0 and 28 respectively. It is also noticeable that the viterbi decoding weight decreases with the MCS, starting at 46 % for MCS 0 and dropping to 12 % when MCS is equal to 28.

IX. Conclusions and Future Work

This paper presented a open source SDR-based framework that enables experimental research and prototyping for various next generation wireless networks spectrum sharing scenarios. Its high configurability, supported by an interface built upon popular programming libraries (Google protobuf and ZeroMQ), allows engineers and researchers to easily

modify it or extend its functions by plugging in new modules implementing novel spectrum sharing techniques and approaches. Additionally, the proposed slot-based PHY is of great importance to combat spectrum scarcity as it has a pivotal role in providing optimum utilization of time-frequency resources. These features make it the perfect candidate for an extensive range of spectrum sharing experimentations in real-world or realistic environments such as testbeds aiming at better understanding disruptive spectrum sharing schemes. Three distinctive use cases are presented with hints on how the proposed framework could be employed in intelligent spectrum sharing research.

As future work, we aim at adding support to self-contained slots and scalable OFDM-based air interface. With self-contained slots, both data and acknowledgement (ACK) information are present in the same time slot. This feature allows each transmission to be a modular transaction, giving PHY the ability to independently decode slots and avoid static timing relationships across slots like in LTE systems. Moreover, self-contained slots are a key enabler to low latency. With a scalable air interface numerology, parameters like subcarrier spacing and transmission time interval (TTI) can be dynamically and efficiently modified to support the diverse frequency, channel bandwidths, deployments, and services foreseen to exist in next generation mobile networks. Additionally, we also plan to offload a few time-consuming PHY processing tasks (e.g., slot synchronization, FIR filtering, channel coding/decoding, and OFDM modulation/demodulation) to the FPGA as a way to increase both real-time and processing performance.

APPENDIX I
EXAMPLE OF A SIMD FIR FILTER

Here in this appendix, we present a example on how to use Advanced Vector Extension (AVX) instructions to implement optimized (in the sense of processing speed) FIR filters. AVX are extensions to the x86 instruction set architecture for microprocessors from Intel and AMD. AVX expands most integer commands to 256 bits and therefore, the registers can store eight 32-bit single-precision floating point numbers or four 64-bit double-precision floating point numbers. We used Intel intrinsic instructions for the FIR filter development. Intel intrinsic instructions are C style functions that provide access to many Intel instructions without the need to write assembly code. A nice interactive reference tool for the intrinsic instructions can be found at [91]. FIR filters are widely employed in digital communications systems, especially in physical layer processing and here, in this work, it is mainly used to implement an filtered version of the OFDM waveform. An L-tap FIR filter can be described as

$$y(n) = \sum_{l=0}^{L-1} a(l)x(n-l), \ \ 0 \leq n \leq L+N-2, \quad (13)$$

where $x(l)$ are the complex input samples, $y(n)$ are the complex output samples and $a(n)$ are the floating point

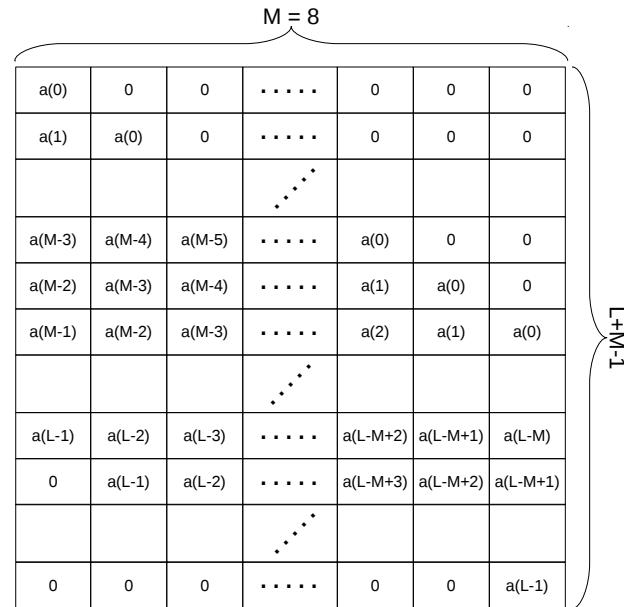



Fig. 29. Memory layout of the FIR filter kernel.

filter coefficients. SIMD instructions allow the processing of multiple samples simultaneous. For instance, SSE instructions support a 256-bit packed vector and therefore, it can process 8 single-precision or 4 double-precision floating values at the same time. Therefore, with AVX functions it is possible to perform $M = 8$ operations at the same time. An important hint on the operation of the intrinsics instructions is that all the data should be aligned for improved performance of the instructions. When using AVX intrinsics, all data should be aligned to a 32-byte memory boundary, i.e., the memory address must be divisible by 32.

In order to speed up the filter processing, the data layout in memory of the filter coefficients should be meticulously designed. Figure 29 depicts the memory layout of the filter kernel (i.e., the FIR filter coefficients). Each row of the filter kernel forms a vector containing M elements for the AVX operations. As can be seen in the figure, the FIR filter coefficients are repeated in each column by applying an offset of one element to each new column. Therefore, the filter kernel is a matrix with dimension $(L + M - 1) \times M$. The filter coefficients are real numbers represented by 1 single-precision floating point value.

The Algorithm 1 shows a exemplary pseudo-code on how a FIR filter can be implemented to take advantage of the AVX functions, and consequently speed-up the filtering time. It receives an input complex array of length N, the filter kernel, which is composed of $(L + M - 1)$ packed vectors with M elements each (i.e., each row is a 256-bit packed vector), and outputs the filtered signal in the output complex array, which will have a length of $L + N - 1$. We assume that the length of the input signal vector is an integer multiple of M, otherwise, it is necessary to pad zeros to the vector until its length becomes an integer multiple of M. Each sample

Algorithm 1 Pseudo-code for an AVX optimized FIR filter.

```
 1: procedure AVXFIRFILTER(cf_t *cplx_in, int in_len, __m256 *fil-
    ter_kernel, int filter_len, cf_t *cplx_out)
 2:     __attribute__((aligned(32))) __m256 blk0, blk1, blkI, blkQ;
 3:     __attribute__((aligned(32))) __m256 dpI, resI, dpQ, resQ;
 4:
 5:     ▷ Static initializations.
 6:     __m256i mask = _mm256_setr_epi32(0, 1, 4, 5, 2, 3, 6, 7);
 7:     const int imm8_I = _MM_SHUFFLE(2, 0, 2, 0);
 8:     const int imm8_Q = _MM_SHUFFLE(3, 1, 3, 1);
 9:     ▷ Cast complex samples into single precision floating-point type.
10:     ▷ Note that one complex sample becomes 2 floating-point values.
11:     __attribute__((aligned (32))) float *in = (float*)cplx_in;
12:     __attribute__((aligned (32))) float *out = (float*)cplx_out;
13:     int cnt = 0, nof_outer_loop_iter = in_len/M;
14:
15:     ▷ Outer loop.
16:     for k ← 0 to (nof_outer_loop_iter - 1) do
17:         ▷ Load 8 values from the input vector.
18:         blk0 = _mm256_load_ps(in+(k*2*M));
19:         ▷ Load the next 8 values from the input vector.
20:         blk1 = _mm256_load_ps(in+(k*2*M)+M);
21:         ▷ Get 8 real values in vector. They are not in time sequence.
22:         blkI = _mm256_shuffle_ps(blk0, blk1, imm8_I);
23:         ▷ Organize real samples in time sequence.
24:         blkI = _mm256_permutevar8x32_ps(blkI, mask);
25:         ▷ Get 8 imag values in vector. They are not in time sequence.
26:         blkQ = _mm256_shuffle_ps(blk0, blk1, imm8_Q);
27:         ▷ Organize imag samples in time sequence.
28:         blkQ = _mm256_permutevar8x32_ps(blkQ, mask);
29:
30:         cnt = 0;
31:         ▷ Inner loop.
32:         for i ← 0 to  (filter_len + M - 2) do
33:             ▷ Multiply lower 4 real elements and sum their products.
34:             ▷ Simulaneously does the same for the higher 4 elements.
35:             dpI = _mm256_dp_ps(filter_kernel[i], blkI, 0xff);
36:             ▷ Swap lower 4 elements with higher 4 elements.
37:             resI = _mm256_permute2f128_ps(dpI, dpI, 1);
38:             ▷ Get the dot product of 2 × eight elements.
39:             resI = _mm256_add_ps(resI, dpI);
40:
41:             ▷ Multiply lower 4 imag elements and sum their products.
42:             ▷ Simulaneously does the same for the higher 4 elements.
43:             dpQ = _mm256_dp_ps(filter_kernel[i], blkQ, 0xff);
44:             ▷ Swap lower 4 elements with higher 4 elements.
45:             resQ = _mm256_permute2f128_ps(dpQ, dpQ, 1);
46:             ▷ Get the dot product of 2 × eight elements.
47:             resQ = _mm256_add_ps(resQ, dpQ);
48:
49:             ▷ Accumulates the results in the output vector.
50:             out[k*2*M + cnt] += resI[0];
51:             out[k*2*M + cnt + 1] += resQ[0];
52:             cnt += 2;
53:         end for
54:     end for
55: end procedure
```

of the input signal is a complex number represented by 2 single-precision floating point values, i.e., real (In-phase) and imaginary (Quadrature) values. This way, when a complex sample vector is cast into a floating-point vector (see lines 11 and 12), the even and odd indexes represent real (I) and imaginary (Q) values respectively.

In the outer loop, after casting the complex input vector into a floating-point vector, in lines 17 to 28, 16 consecutive floating-point values are rearranged into two 256-bit packed vectors, the first one containing only real values and the second

TABLE X

FIR FILTER PROCESSING TIME FOR DIFFERENT IMPLEMENTATION VERSIONS.

FIR order	FIR version	Time [ms]	Speed-up %
64	naive	0.954478	-
	no unrolling	0.133955	85.966
	unrolling	0.093086	90.247
128	naive	1.986	-
	no unrolling	0.234	88.211
	unrolling	0.197	90.090

one containing only imaginary values. At each iteration of the outer loop, 8 real and 8 imaginary values are loaded into 256-bit packed vectors. In the inner loop, the the dot product between each one of the rows of the filter kernel and the real packed vector is calculated from lines 33 to 39 and the result is accumulated to the even indexed output vector, in line 50. The the dot product between each one of the rows of the filter kernel and the imaginary packed vector is calculated from lines 41 to 47 and the result is accumulated to the add indexed output vector, in line 51. Therefore, at each iteration of the inner loop two, real and imaginary, output values are calculated and accumulated in the output vector. The output complex vector should be zeroed before the filtering as we use its contents to accumulate the output values.

It is worth mentioning that the filter processing time can be further decreased by unrolling the inner-loop, which is done by replicating lines 33 to 52 for subsequent rows of the filter kernel.

In Table X we compare the processing time for different versions of the FIR filter implementation. For this comparison we use an input signal with 5760 complex samples and FIR filters with length equal to 65 and 129 respectively, i.e., 64-th and 128-th order FIR filters. The FIR processing time is averaged over 100×10^3 repetitions of the filtering operation. The *naive* version corresponds to the straight-forward implementation of (13), the *no unrolling* version corresponds to the pseudo-code presented in Algorithm 1 and the *unrolling* version corresponds to unrolling the inner-loop 8 times.

APPENDIX II
SUBBAND OCCUPANCY ALGORITHM

In this section we describe the Subband Occupancy Algorithm in details. As mentioned earlier, the algorithm is based on the Cell-Average Constant False Alarm Rate (CA-CFAR) algorithm [62]. It is employed for detecting the presence of transmissions, i.e., occupancy, of subbands composing the bandwidth being assessed. The algorithm is used to calculate an occupancy (i.e., detection) threshold based on a reference set of subband power values. The algorithm can be split into two three stages: (i) removal of samples that are not considered as only being noise observations, this step is also known as censoring, (ii) calculation of a noise-power threshold, and (iii) the actual subband occupancy detection. This procedure makes it possible to reliably decide whether a subband, sb, is already

being occupied by another user or not. Figure 30 summarizes the three stages composing the Subband Occupancy algorithm.

The sample power calculation and subband grouping processing tasks are realized as showed in Figure 10 and described in Section V. The subsequent signal processing is performed in three stages making use of the subband power samples. The subband occupancy procedure consists basically of hypothesis testing following the Neyman&Pearson lemma [92]. This lemma establishes that subband occupancy testing is based on the following likelihood ratio inequality:

$$\frac{sb^{H_1}}{sb^{H_0}} > \alpha, \tag{14}$$

where the hypothesis H_0 is rejected in favor of H_1 when the subband contains not only noise, i.e., the presence of a transmission. This hypothesis testing is optimum only when the cumulative distribution function (CDF) of the ratio in (14) given the hypothesis H_0 is known, so that it is possible to calculate a threshold value that satisfies:

$$P\left\{\frac{sb^{H_1}}{sb^{H_0}} > \alpha \mid H_0\right\} = P_{FR}, \tag{15}$$

for a given probability of false rejection, P_{FR}. Therefore, normally, the derivation of the threshold assumes knowledge of the probability distribution function (PDF) of both random variables sb^{H_1} and sb^{H_0}.

A. Censoring and Noise Power Reference, Z_{REF} Calculation Stage

In this stage, corrupted subband power samples are removed from the reference set. Corrupted subband power samples are caused by the presence of transmissions in that section (subband) of the assessed spectrum band. For the censoring of subband power samples, we use the Forward Consecutive Mean Excision (FCME) algorithm [93]. Initially, the values in the subband power reference set, defined as

$$\{sb(l) \mid l = 0, 1, \cdots, M-1\}, \tag{16}$$

are sorted in ascending order, so that

$$\left\{sb^{(i)} \mid i = 0, 1, \cdots, M-1\right\}, \tag{17}$$

where

$$sb^{(0)} \leq sb^{(1)} \leq sb^{(2)} \leq \cdots \leq sb^{(M-1)}. \tag{18}$$

The censoring at the k-th step is performed with the following inequality

$$sb^{(k+1)} \geq T_k \sum_{i=1}^{k} sb^{(i)}, \tag{19}$$

where T_k is the censoring scaling factor at the k-th step. If the inequality is satisfied (i.e., true), all the greater values are decided to be corrupted by the presence/occupancy of transmissions/interference from other users. The test of (19) is performed iteratively, being necessary to calculate the censoring scaling factor, T_k, for each new iteration. The censoring scaling factor, T_k, controls the properties of the censoring

stage, and it is normally chosen so that the probability of false disposal, P_{FD}, has a pre-defined value. P_{FD} is the probability of making a incorrect decision during the k-th step of the censoring stage. The initial T_k value is calculated assuming that $sb^{(k+1)}$ is a subband power sample that only contains noise observation, i.e., it is free of transmission/interference signals. Under this assumption, the probability that the inequality in (19) is true corresponds to a probability of false disposal, P_{FD} defined by

$$P_{FD} = P\left\{sb^{(k+1)} \geq T_k \sum_{i=1}^{k} sb^{(i)} \mid H_0\right\}. \tag{20}$$

Each iteration of the censoring stage starts with k equal to the smallest assumed clean set of subbands samples. The larger the smallest assumed clean set of subbands samples, the better the censoring algorithm works [93]. However, the larger the assumed clean set of subbands samples, the higher is the probability that corrupted subband samples will be part of the initial clean set. The testing of (19) continues until it is true for some value of k or all reference subband power samples are decided to be unoccupied, i.e., free of transmissions/interference.

Considering the H_0 hypothesis (i.e., only noise observations) and that noise follows a complex Gaussian distribution, $\mathcal{CN}(0, \sigma^2)$, the noise distribution after the N-point FFT also follows a complex Gaussian distribution (the only difference is a scaling factor, N, due to the FFT processing), defined by $\mathcal{CN}(0, N\sigma^2)$. Therefore, each one of the subband power values, $sb(i) = \sum_{b=0}^{B-1} |x[b]|^2, i = 0, 1, \cdots, M-1$, follows a Gamma distribution, $\Gamma(B, N\sigma^2)$ with mean given by

$$\mathbb{E}[sb^{(\cdot)}] = BN\sigma^2. \tag{21}$$

Therefore, since that

$$\frac{\sum_{i=0}^{k-1} sb^{(i)}}{k} \approx \mathbb{E}[sb^{(\cdot)}] = BN\sigma^2, \tag{22}$$

the equation for the probability of false disposal, P_{FD}, can be approximated as

$$P_{FD} \approx P\left\{sb^{(k+1)} \geq T_k k BN\sigma^2\right\}. \tag{23}$$

This approximation becomes better as the number of reference subband power samples, k, increases. Additionally, given that $sb^{(k+1)}$ is a Gamma random variable, $\Gamma(k, \theta)$, with shape parameter being a positive integer number, B, its cumulative distribution function (CDF) can be simplified and expressed in closed form as [94, 95]

$$F(y, k, \theta) = P(Y \leq y) = 1 - e^{-y/\theta} \sum_{i=0}^{k-1} \frac{1}{i!} \left(\frac{y}{\theta}\right)^i, \tag{24}$$

where k and θ are the shape and scale parameters respectively.

Therefore, the probability of false discard (or censoring) in (23) can be calculated as

$$P_{FD} = 1 - P\left\{sb^{(k+1)} \leq T_k k BN\sigma^2\right\}$$
$$= e^{-T_k kB} \sum_{i=0}^{B-1} \frac{1}{i!} (T_k kB)^i, \tag{25}$$

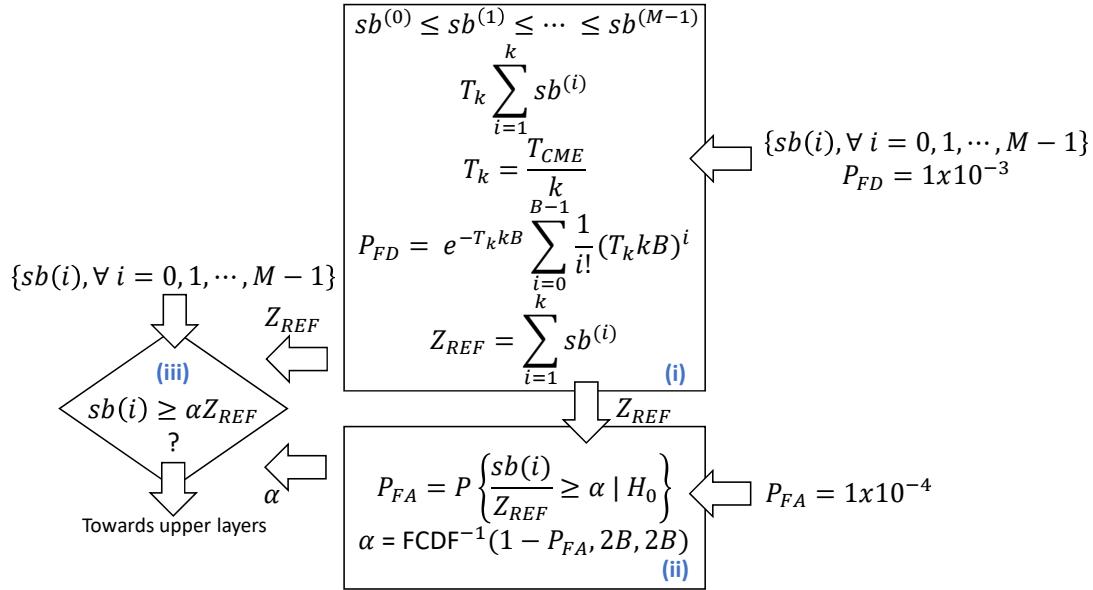



Fig. 30. Summary of the subband occupancy algorithm: stage (i) is used to calculate the noise-power reference, Z_{REF}; stage (ii) is used to calculate the threshold, α; and stage (iii) applies hypothesis test to all the subband power values, $sb(i)$.

where in (24), y was made equal to $T_k k B N sigma^2$. Here we define $T_{CME} = kT_k$, which represents a input parameter for the FCME algorithm and can be found by solving (25). P_{FD} can be understood as the desired clean sample rejection rate.

The censoring threshold is defined as $T_k \sum_{i=1}^{k} sb^{(i)}$ and therefore, subband power samples that have value greater or equal than the threshold are censored (i.e., discarded) and the new set of subband power samples consists of the remaining samples, which are assumed to contain only noise observations. The output of the disposal, i.e., censoring, stage are composed by the number of clean subband power samples, k, and the *noise-power reference*, Z_{REF}, which is defined by the summation of all subband power samples that have value below the censoring threshold. Z_{REF} is defined as

$$Z_{REF} = \sum_{i=1}^{k} sb^{(i)}. \qquad (26)$$

B. Threshold Calculation Stage

After selecting the set of subband power samples that are considered as only being noise observations, the next stage consists of calculating the occupancy (or detection) threshold, α. The occupancy threshold, α, is calculated according to the employed decision method, which here is selected as being the Cell Averaging (CA) method [93].

The occupancy threshold, α, is calculated under the assumption that there is no transmission/interference present in the reference subband power samples for a given probability of false alarm, P_{FA}, which is defined as

$$P_{FA} = P\left\{ \frac{sb(i)}{Z_{REF}} \geq \alpha \mid H_0 \right\}. \qquad (27)$$

As the quadrature components (i.e., complex samples) at the output of the FFT are assumed to follow the complex

Gaussian distribution, $\mathcal{CN}(0, N\sigma^2)$, it can be consequently assumed that the subband power samples $sb(.)$ and the noise-power reference, Z_{REF}, follow scaled central Chi-squared distributions, $\frac{N\sigma^2}{2}\chi^2(2B)$, $\frac{N\sigma^2}{2}\chi^2(2Bk)$ with $2B$ and $2Bk$ degrees of freedom, respectively. Additionally, it is known that the ratio between two central Chi-squared random variables results in a random variable that follows a *Fisher* distribution [93, 96], whose CDF is defined as

$$\text{FCDF}(\alpha) = 1 - P\left\{ \frac{sb(i)/2B}{Z_{REF}/2Bk} \geq k\alpha \right\}, \qquad (28)$$

therefore, the occupancy threshold, α, is calculated as

$$\alpha = \text{FCDF}^{-1}(1 - P_{FA}, 2B, 2Bk)/k. \qquad (29)$$

where FCDF is the *Fisher* CDF. Note that the scaling factor, $\frac{N\sigma^2}{2}$, is the same for $sb(.)$ and Z_{REF}, and therefore, it disappears when the ratio is taken.

C. Subband Occupancy Testing Stage

After calculating the occupancy threshold, α, and the noise-power reference, Z_{REF}, the final decision if a subband is free or occupied is made by evaluating the following hypothesis test

$$sb(i) \geq \alpha Z_{REF}, \forall \, i = 0, 1, \cdots, M - 1. \qquad (30)$$

If the test is true, signal(s)-plus noise hypothesis H_1 is chosen, i.e., the subband is occupied by a transmission. Otherwise, the noise-only hypothesis H_0 is decided to be true and the subband is declared unoccupied.

ACKNOWLEDGEMENTS

The project leading to this publication has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 732174 (ORCA project).

REFERENCES

[1] Cisco, *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2015-2020*, White paper, February 2016.

[2] The 5G Infrastructure Public Private Partnership (5G-PPP) Technical Report, *5G empowering vertical industries*, February 2016.

[3] 3GPP Technical Report TR 38.913, *Study on scenarios and requirements for next generation access technologies, version 14.2.0, Release 14*, March 2017.

[4] Michael Koziol, *5G New Radio and What Comes Next*, IEEE Spectrum, January 2018.

[5] Gregory Staple and Kevin Werbach, *The End of Spectrum Scarcity*, IEEE Spectrum, March 2014.

[6] Recommendation ITU-R M.2083, *IMT Vision - Framework and overall objectives of the future development of IMT for 2020 and beyond*, September 2015.

[7] ITU-R Report M.2370-0, *IMT traffic estimates for the years 2020 to 2030*, July 2015.

[8] Keith Mallinson, *The path to 5G: as much evolution as revolution*, May 2016. [available at] http://www.3gpp.org/news-events/3gpp-news/1774-5g_wiseharbour.

[9] Oumer Teyeb, Gustav Wikstrom, Magnus Stattin, Thomas Cheng, Sebastian Faxer, and Hieu Do, *Evolving LTE to fit the 5G future*, Ericsson Technology Review, January 2017. [available at] https://www.ericsson.com/en/ericsson-technology-review/archive/2017/evolving-lte-to-fit-the-5g-future

[10] Konstantinos Chatzikokolakis, Panagiotis Spapis, Alexandros Kaloxylos, and Nancy Alonistioti, *Toward spectrum sharing: opportunities and technical enablers*, IEEE Communications Magazine, vol. 53, no. 7, July 2015.

[11] Junyu Liu, Min Sheng, Lei Liu, and Jiandong Li, *Network Densification in 5G: From the Short-Range Communications Perspective*, IEEE Communications Magazine, vol. 55, no. 12, December 2017.

[12] Yan Li, and Shaoyi Xu, *Traffic Offloading in Unlicensed Spectrum for 5G Cellular Network: A Two-Layer Game Approach*, MDPI Entropy, January 2018.

[13] Federal Communications Commission, *Spectrum Policy Task Force Report*, [online]. Available: https://www.fcc.gov/document/spectrum-policy-task-force, FCC, November 2002.

[14] Mina Labib, Vuk Marojevic, Jeffrey H. Reed, and Amir I. Zaghloul, *Extending LTE into the Unlicensed Spectrum: Technical Analysis of the Proposed Variants*, IEEE Communications Standards Magazine, vol. 1, no. 4, December 2017.

[15] Qualcomm Technologies, Inc., *LTE-U/LAA, MuLTEfire and Wi-Fi; making best use of unlicensed spectrum*, White Paper, September 2015.

[16] MulteFire Alliance, *MulteFire Release 1.0 Technical Paper: A new way to wireless*, January 2017.

[17] Andreas Roessler, *Impact of spectrum sharing on 4G and 5G standards a review of how coexistence and spectrum sharing is shaping 3GPP standards*, IEEE International Symposium on Electromagnetic Compatibility & Signal/Power Integrity (EMCSI), October 2017.

[18] Ahsan Saadat, Gengfa Fang, and Wei Ni, *A Two-Tier Evolutionary Game Theoretic Approach to Dynamic Spectrum Sharing through Licensed Shared Access*, IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing, October 2015.

[19] Nadine Abbas, Youssef Nasser, and Karim El Ahmad, *Recent advances on artificial intelligence and learning techniques in cognitive radio networks*, EURASIP Journal on Wireless Communications and Networking, vol. 2015, no. 1, pp. 174-194, December 2015.

[20] Seppo Yrjola, and Heikki Kokkinen, *Licensed Shared Access evolution enables early access to 5G spectrum and novel use cases*, EAI Endorsed Transactions on Wireless Spectrum, December 2017.

[21] J. Mitola, *The software radio architecture*, IEEE Communications Magazine, vol. 33, no. 5, May 1995.

[22] J. Mitola, *Software radios-survey, critical evaluation and future directions*, National Telesystems Conference (NTC), May 1992.

[23] Felipe A. P. de Figueiredo, *A Framework for Intelligent Spectrum Sensing and Sharing*, [online]. Available: https://github.com/zz4fap/intelligent-spectrum-sharing, May 2018.

[24] Felipe A. P. de Figueiredo, Xianjun Jiao, Wei Liu, Irfan Jabandzic, Spilios Giannoulis, and Ingrid Moerman, *A Framework for Intelligent Spectrum Sharing*, Preprints 2018, 2018080447 (doi: 10.20944/preprints201808.0447.v1).

[25] Ettus Research LLC, *USRP Hardware Driver*, [online]. Available: http://files.ettus.com/manual/

[26] Ettus Research LLC, *Products*, [online]. Available: https://www.ettus.com/product

[27] *Open Air Interface*, 2015. [online]. Available: http://www.openairinterface.org/

[28] *openLTE*, 2015. [online]. Available: http://openlte.sourceforge.net/

[29] I. Gomez-Miguelez, A. Garcia-Saavedra, P. D. Sutton, P. Serrano, C. Cano and D. J. Leith, *srsLTE: An Open-Source Platform for LTE Evolution and Experimentation*, ACM WiNTECH Workshop, October 2016.

[30] Amarisoft Corp. *Amari LTE 100 - Software LTE base station on PC*, [online]. Available: http://www.amarisoft.com

[31] National Instruments, *LabVIEW Communications LTE Application Framework 2.0 and 2.0.1*, White Paper, December 2016. [online]. Available: http://www.ni.com/white-paper/53286/en/

[32] National Instruments, *Real-time LTE/Wi-Fi Coexistence Testbed*, White Paper, February 2016. [online]. Available: http://www.ni.com/white-paper/53044/en/

[33] Zhenyu Zhou, Yunjian Jia, Fei Chen, Kimfung Tsang, Guangyi Liu, and Zhu Han, *Unlicensed Spectrum Sharing: From Coexistence to Convergence*, IEEE Wireless Communications, vol. 24, no. 5, October 2017.

[34] Cristina Cano, Douglas J. Leith, Andres Garcia-Saavedra, and Pablo Serrano, *Fair Coexistence of Scheduled and Random Access Wireless Networks: Unlicensed LTE/Wi-Fi*, IEEE/ACM Transactions on Networking, vol. 25, no. 6, December 2017.

[35] D. Ruby, M. Vijayalakshmi, and A. Kannan, *Intelligent relay selection and spectrum sharing techniques for cognitive radio networks*, Cluster Computing, August 2017.

[36] Usama Mir, Leila Merghem-Boulahia, Dominique Gaiti, *Dynamic Spectrum Sharing in Cognitive Radio Networks: a Solution based on Multiagent Systems*, International Journal on Advances in Telecommunications, vol. 3 no. 3, January 2010.

[37] Nadisanka Rupasinghe, and Ismail Guvenc, *Reinforcement learning for licensed-assisted access of LTE in the unlicensed spectrum*, IEEE Wireless Communications and Networking Conference (WCNC), June 2015.

[38] Ruben Mennes, Miguel Camelo, Maxim Claeys and Steven Latre, *A Neural-Network-based MF-TDMA MAC Scheduler for Collaborative Wireless Networks*, IEEE Wireless Communications and Networking Conference (WCNC), April 2018.

[39] Zoraze Ali, Lorenza Giupponi, Josep Mangues-Bafalluy, and Biljana Bojovic, *Machine learning based scheme for contention window size adaptation in LTE-LAA*, IEEE Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), February 2018.

[40] P. Hintjens, *ZeroMQ Messaging for Many Applications*, O'Reilly Media, March 2013.

[41] Ettus Research LLC, *USRP X Series*, [online]. Available: https://www.ettus.com/product/category/USRP-X-Series

[42] National Instruments, *Overview of the NI USRP RIO Software Defined Radio*, White Paper, December 2015. [online]. Available: http://www.ni.com/white-paper/52119/en/

[43] Google, *protobuf - Protocol Buffers - Google's data interchange format*, [online]. Available: http://code.google.com/p/protobuf/, 2011.

[44] DARPA SC2 Website: https://spectrumcollaborationchallenge.com/.

[45] DAPRA, *Kick-off meeting*, The Johns Hopkins University, January 2017.

[46] Cristina Cano, and Douglas J. Leith, *Unlicensed LTE/Wi-Fi coexistence: Is LBT inherently fairer than CSAT?*, IEEE International Conference on Communications (ICC), July 2016.

[47] Houman Zarrinkoub, *Overview of the LTE Physical Layer*, John Wiley & Sons, January 2014.

[48] Ali A. Zaidi, Robert Baldemair, Hugo Tullberg, Hakan Bjorkegren, Lars Sundström, Jonas Medbo, Caner Kilinc, and Icaro Da Silva, *Waveform and Numerology to Support 5G Services and Requirements*, IEEE Communications Magazine, vol. 54, no. 11, pp. 90-98, November 2016.

[49] Sravanthi Kanchi, Shubhrika Sandilya, Deesha Bhosale, Adwait Pitkar, and Mayur Gondhalekar, *Overview of LTE-A technology*, IEEE Global High Tech Congress on Electronics (GHTCE), March 2014.

[50] Shao-Yu Lien, Shin-Lin Shieh, Yenming Huang, Borching Su, Yung-Lin Hsu, and Hung-Yu Wei, *5G New Radio: Waveform, Frame Structure, Multiple Access, and Initial Access*, IEEE Communications Magazine, vol. 55, no. 6, June 2017.

[51] Naga Bhushan, Tingfang Ji, Ozge Koymen, John Smee, Joseph Soriaga, Sundar Subramanian, and Yongbin Wei, *5G Air Interface System Design Principles*, IEEE Wireless Communications, October 2017.

[52] ANSI/IEEE 802.11n-2009 , *IEEE Standard for Information technology - Local and metropolitan area networks - Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC)and Physical Layer (PHY) Specifications Amendment 5: Enhancements for Higher Throughput*, 2010.

[53] ETSI TR 101.190, *Digital Video Broadcasting (DVB); Framing structure, channel coding and modulation for digital terrestrial television - V1.6.1*, January 2009.

[54] Feng Wang, and Yu Zhu, *An efficient CFO estimation algorithm for the downlink of 3GPP-LTE*, International Conference on Wireless Communications and Signal Processing (WCSP), November 2011.

[55] Gwanmo Ku, and John MacLaren Walsh, *Resource Allocation and Link Adaptation in LTE and LTE Advanced: A Tutorial*, IEEE Communications Surveys & Tutorials, vol. 17, no. 3, December 2014.

[56] 3GPP TS 36.213, *Physical Layer Procedures*, 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA). URL: http://www.3gpp.org, May 2016.

[57] Quentin Bodinier, Faouzi Bader, and Jacques Palicot, *On Spectral Coexistence of CP-OFDM and FB-MC Waveforms in 5G Networks*, IEEE Access, vol. 5, July 2017.

[58] Hao Lin, and Pierre Siohan, *Major 5G Waveform Candidates: Overview and Comparison*, Book chapter in Signal Processing for 5G, Wiley, August 2016.

[59] Lei Zhang, Ayesha Ijaz, Pei Xiao, Mehdi M. Molu, and Rahim Tafazolli, emphFiltered OFDM Systems, Algorithms, and Performance Analysis for 5G and Beyond, IEEE Transactions on Communications, vol. 66, no. 3, March 2018.

[60] Javad Abdoli, Ming Jia, and Jianglei Ma, *Filtered OFDM: A New Waveform for Future Wireless Systems*, IEEE International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), August 2015.

[61] Xi Zhang, Ming Jia, Lei Chen, Jianglei Ma, and Jing Qiu, *Filtered-OFDM - Enabler for Flexible Waveform in the 5th Generation Cellular Networks*, IEEE Global Communications Conference (GLOBECOM), December 2015.

[62] Felipe A. P. de Figueiredo, Fabbryccio A. C. M. Cardoso, Jose A. Bianco F., Rafael M. Vilela, Karlo G. Lenzi, *Multi-stage Based Cross-Correlation Peak Detection for LTE Random Access Preambles*, Revista Telecomunicacoes, vol. 15, September 2013.

[63] Janne J. Lehtomaki, Johanna Vartiainen, Markku Juntti and Harri Saarnisaari, *Spectrum sensing with forward methods*, in Proc. IEEE Military Commun. Conf., pp. 1-7, Washington, D.C., USA, October 2006.

[64] Gen Li, Tan Wang, Qingyu Miao, Ying Wang, and Biao Huang, *Spectrum Sharing for 5G*, 5G Mobile Communications, Springer, Cham, October 2016.

[65] X. Wang et al., *Throughput and Fairness Analysis of Wi-Fi and LTE-U in Unlicensed Band*, IEEE JSAC, vol. 35, no. 1, pp. 63-78, January 2017.

[66] 3GPP TR 36.889, *Feasibility Study on Licensed-Assisted Access to Unlicensed Spectrum (Release 13)*, January 2015.

[67] Mina Labib, Vuk Marojevic, Jeffrey H. Reed, and Amir I. Zaghloul, *Extending LTE into the Unlicensed Spectrum: Technical Analysis of the Proposed Variants*, IEEE Communications Standards Magazine, vol. 1, no. 4, December 2017.

[68] Jaehong Yi, Weiping Sun, Seungkeun Park, and Sunghyun Choi, *Performance Analysis of LTE-LAA Network*, IEEE Communications Letters, December 2017.

[69] Xianjun Jiao, Ingrid Moerman, Wei Liu, and Felipe A. P. de Figueiredo, *Radio Hardware Virtualization for Coping with Dynamic Heterogeneous Wireless Environments*, International Conference on Cognitive Radio Oriented Wireless Networks (CROWNCOM), September 2017.

[70] Felipe A. P. de Figueiredo, Xianjun Jiao, Wei Liu, and Ingrid Moerman, *Radio Hardware Virtualization for Software-Defined Wireless Networks*, Wireless Personal Communications, vol. 100, no. 1, pp. 113-126, March 2018.

[71] Vasilis Maglogiannis, Dries Naudts, Adnan Shahid, and Ingrid Moerman, *An adaptive LTE listen-before-talk scheme towards a fair coexistence with Wi-Fi in unlicensed spectrum*, Telecommunication Systems, January 2018.

[72] Anand M. Baswade, and Bheemarjuna Reddy Tamma, *Channel sensing based dynamic adjustment of contention window in LAA-LTE networks*, International Conference on Communication Systems and Networks (COMSNETS), March 2016.

[73] Qualcomm Technologies, Inc., *What can we do with 5G NR Spectrum Sharing that isn't possible today?*, White Paper, December 2017.

[74] Ericsson AB, *Spectrum sharing* White Paper, October 2013.

[75] Hai-Yan Shi, Wan-Liang Wang, Ngai-Ming Kwok, and Sheng-Yong Chen, *Game Theory for Wireless Sensor Networks: A Survey*, MDPI Sensors, vol. 12, no. 7, July 2012.

[76] Saed Alrabaee, Mahmoud Khasawneh, Anjali Agarwal, Nishith Goel, and Marzia Zaman, *A game theory approach: Dynamic behaviours for spectrum management in cognitive radio network*, IEEE Globecom Workshops (GC Wkshps), December 2012.

[77] Prabodini Semasinghe, Setareh Maghsudi, and Ekram Hossain, *Game Theoretic Mechanisms for Resource Management in Massive Wireless IoT Systems*, IEE Communications Magazine, vol. 55, no. 2, February 2017.

[78] Ettus Research LLC, *CBX 1200-6000 MHz Rx/Tx (120 MHz, X Series only)*, [online]. Available: https://www.ettus.com/product/details/CBX120

[79] 3GPP TS 36.104, *Base Station (BS) radio transmission and reception - V14.3.0*, 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA). URL: http://www.3gpp.org, April 2017.

[80] 3GPP TS 36.141, *Base Station (BS) Conformance Testing - V8.4.0*, 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA). URL: http://www.3gpp.org, September 2009.

[81] 3GPP TS 36.521-1, *User Equipment (UE) Conformance Specification; Radio Transmission and Reception Part 1: Conformance Testing - V8.3.1*, 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA). URL: http://www.3gpp.org, September 2009.

[82] Mohammad T. Kawser, Nafiz Imtiaz Bin Hamid, Md. Nayeemul Hasan, M. Shah Alam, and M. Musfiqur Rahman, *Downlink SNR to CQI Mapping for Different Multiple Antenna Techniques in LTE*, International Journal of Information and Electronics Engineering, vol. 2, no. 5, September 2012.

[83] 3GPP TS 36.141. *Base Station (BS) conformance testing*, 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA). URL: http://www.3gpp.org.

[84] Van de Beek, J. J., O. Edfors, M. Sandell, S. K. Wilson, and P. O. Borjesson, *On Channel Estimation in OFDM Systems*, IEEE Vehicular Technology Conference, vol. 2, 1995.

[85] Kazuki Takeda, Li Hui Wang, and Satoshi Nagata, *Latency reduction towards 5G*, IEEE Wireless Communications, vol. 24, no. 3, pp. 2-4, June 2017.

[86] D. Qijia, Z. Jun, Z. Tao, and Q. Yong, *Resource Allocation Strategies in MF-TDMA Satellite Systems*, Acta Aeronautica Et Astronautica Sinica, vol. 30 no. 9, pp. 1718-1726, September 2009.

[87] *psutil*, [online]. Available: https://pypi.org/project/psutil/

[88] Jose Mendes, et al., *Cellular Access Multi-Tenancy through Small Cell Virtualization and Common RF Front-End Sharing*, Proceedings of the 11th Workshop on Wireless Network Testbeds, Experimental evaluation & Characterization, October 2017.

[89] Josef Weidendorfer, Markus Kowarschik, and Carsten Trinitis, *A Tool Suite for Simulation Based Analysis of Memory Access Behavior*, Proceedings of the 4th International Conference on Computational Science (ICCS), Krakow, Poland, June 2004.

[90] Julian Seward, and Nicholas Nethercote, *Using Valgrind to detect undefined value errors with bit-precision*, Proceedings of the USENIX'05 Annual Technical Conference, Anaheim, California, USA, April 2005.

[91] Intel Corporation, *Intel's Intrinsics Guide*, [online]. Available: https://software.intel.com/sites/landingpage/IntrinsicsGuide/

[92] J. Neyman and E. S. Pearson, *On the problem of the most efficient tests of statistical hypotheses*, Philosophical Transactions of the Royal Society of London, Series A. 231: 289-337, 1933.

[93]   Janne J. Lehtomaki, Markku Juntti and Harri Saarnisaari, *CFAR Strategies for Channelized Radiometer*, IEEE Signal Processing Letters, vol. 12, no. 1, January, 2005.

[94]   J. G. Proakis, *Digital Communications*, 5th Edition, McGraw-Hill, New York, 2001.

[95]   Papoulis, Pillai, *Probability, Random Variables, and Stochastic Processes*, 4th Edition, McGraw-Hill Europe, January 2002.

[96]   M. H. DeGroot, *Probability and Statistics*, 2nd Edition, Addison-Wesley, 1986.