

# *Kitsune*: A Management System for Cognitive Radio Networks Based on Spectrum Sensing

Lucas Bondan, Marcelo Antonio Marotta, Maicon Kist, Leonardo Roveda Faganello,  
Cristiano Bonato Both, Juergen Rochol, Lisandro Zambenedetti Granville  
Institute of Informatics – Federal University of Rio Grande do Sul  
Av. Bento Gonçalves, 9500 – Porto Alegre, Brazil  
Email: {lbondan, mamarotta, maicon.kist, lrfaganello, cbboth, juergen, granville}@inf.ufrgs.br

**Abstract**—Software defined radio enables the improvement of the radio-frequency spectrum utilization through the design of cognitive radio devices. The implementation of these devices must be based on spectrum sensing function searching for vacant channels and, opportunistically, transmit over these channels in a cognitive radio network. Therefore, the configuration, monitoring and visualization of the spectrum sensing function are fundamentals to the continuous learning process of the network administrator. In this paper we propose *Kitsune*, a management system based on a hierarchical model allowing to manage summarized information about the spectrum sensing function in a cognitive radio networks. Moreover, a *Kitsune* prototype was developed and evaluated through a real IEEE 802.22 scenario using TV channels to Internet access. Results shown that *Kitsune* allows network administrator to achieve a higher knowledge about behavior of the users and improve the average throughput for each channel.

**Index Terms**—software defined radio, cognitive radio, network management, spectrum sensing

## I. INTRODUCTION

Cognitive Radio (CR) is an emerging technology, based on Software Defined Radio (SDR), that enables the improvement of the Radio-Frequency (RF) spectrum usage [1]. CR devices can analyze the RF spectrum, searching for channels that are not being used and opportunistically transmit over these channels. To allow opportunistic transmissions, a CR device needs to implement four main functions: spectrum sensing, spectrum decision, spectrum sharing, and spectrum mobility [2]. Among these functions, the spectrum sensing is considered the most important because it is responsible for analyzing the RF spectrum and evaluate whether or not a channel is vacant [3]. Moreover, the behavior of users directly impact in the spectrum sensing evaluation, since these users may transmit in the sensed channel.

The spectrum sensing function must be implemented in each device of a CR network. As defined by the IEEE 802.22 standard, a CR network is composed of two types of devices: Base Stations (BS) and Customer-Premises Equipment (CPE) [4]. In these networks, the spectrum sensing function can be dynamically configured to minimize the sensing time and maximizing both the transmission time and the reliability of spectrum sensing results. This configuration contains a set of parameters, such as the number of sensed channels, sensing

time, and sensing window [5]. The dynamic configuration of these parameters can be improved through a management system. In such system the spectrum sensing function can be monitored and visualized, allowing the administrator to continuously learn how to properly configure the spectrum sensing function of each CR to obtain the best network performance.

In the last decade, much has been discussed about CR devices and networks [6], but there are few discussions about the management of CR networks. This lack of discussions may be due to the fact that the deployment of CR networks has begun only in the last years [7]. Therefore, CR network management has the same classical problem of traditional networks management, *i.e.*, when a new technology is introduced, the management is usually neglected, until eventually it becomes necessary. Currently, researches on CR networks management mainly addressed the usage of management protocols [8], autonomic systems [9], policies [10], spectrum visualization [11], and machine learning algorithms [2]. However, to the best of our knowledge, no management system for CR networks based on the spectrum sensing function has been proposed. Such management system should offer summarized information about the spectrum sensing function, retrieved from each CR device, allowing the administrators to acquire better knowledge about the managed CR network.

In this paper we propose *Kitsune*, a management system for CR networks designed to enable the administrator to continuously learn how to better configure the spectrum sensing function. Management through *Kitsune* is separated into three stages: configuration, monitoring, and visualization. *Kitsune* system follows a hierarchical management model, composed of a Manager, Gateways, and Agents. The main advantage of the proposed system is to enable dynamic configuration and monitoring through an intuitive visualization interface. The *Kitsune* management system is evaluated using a prototype implemented in a experimental scenario based on the IEEE 802.22 standard. We demonstrated that the proposed system enables the network administrator to achieve a better knowledge about the network, improving the throughput obtained in the transmissions performed by the CPEs.

The rest of this paper is organized as follows. In Section II, we present a background on CR network management and

related work in the area. In the Section III, the *Kitsune* management system is explained in details. In the Section IV, the evaluation methodology is explained and results obtained are discussed. Finally, conclusions and future work are provided in Section V.

## II. BACKGROUND ON COGNITIVE RADIO MANAGEMENT

In this section, we present a brief background and related work on CR and its management. In the Subsection II-A, the four main cognitive functions, the traditional hierarchical management model, and the Management Information Base (MIB) of the IEEE 802.22 standard are presented. Afterwards, the related work on CR networks management is discussed in the Subsection II-B.

### A. Cognitive Radio and Network Management

CR devices are foreseen as the future of radio devices, greatly improving the utilization of the RF spectrum. In a CR, almost every aspect of its operation can be dynamically configured [1]. The most important characteristic of the CR is the ability to search the RF spectrum for vacant channels and learn about the RF environment to opportunistically perform transmission over these channels. The literature in CR separates the functionality of these devices in four functions: spectrum sensing, spectrum decision, spectrum sharing and spectrum mobility [12]. Through the spectrum sensing analysis, the CR can identify and use the vacant channels to transmit [3]. Spectrum decision refers to the ability of selecting the best vacant channel, according to some criteria, such as historical availability, channel quality, among others [5]. Spectrum sharing comprises the coordinated access to a vacant channel by two or more CR devices [13]. Finally, spectrum mobility function enables a CR device to leave the channel when another transmission is detected [14].

Among the four functions of a CR device, the spectrum sensing is considered the most important [3]. The main reason for this is because all the other functions can use the spectrum sensing results as input to perform their operations [3]. In addition, the spectrum sensing capability to analyze the RF spectrum and evaluate whether or not a channel is vacant is fundamental to the proper operation of the CR network. Therefore, considering the spectrum sensing importance in the design of CR devices, such as BSs and CPEs, we argue that a CR network management system must be based on the spectrum sensing function.

When a CR network comprises a large number of BSs and CPEs, its management may become complex, given that each BS and CPE can be dynamically configured, monitored and visualized. To facilitate the management of such complex systems, three management entities may be used according to a hierarchical model: (i) Managers are responsible for the management tasks, such as controlling the transmission of each CR device in the network or generating summarized reports about the overall network, (ii) Gateways are capable of executing ordinary actions, such as collecting information about the CR devices in the network and (iii) Agents are placed within

CR devices and are responsible for responding the requests performed by Gateways and applying new configurations to these devices. In the literature, the Agent usually operates the IEEE 802.22 MIB [4], as explained below.

The IEEE 802.22 Standard specifies a MIB module [4] which provides the most relevant information for CR networks management systems. The MIB module is organized in seven groups. These groups are presented in Fig. 1. The *wranDevMib* group informs the software version of the device and which SNMP traps can be configured. The *wranIfBsMib* group presents objects related to the BS operation. The *wranIfBsSfMgmt* group refers to the management items associated with service flow configuration, instantiation, and management. The *wranIfCpeMib* group has objects related to operation of CPEs. The importance assigned to the spectrum sensing function is noticeable due to the *wranIfSmMib* and the *wranIfSsaMib* groups. The first group presents objects associated with the spectrum management, while the second group offers objects related to the spectrum sensing function management. Finally, the *wranIfDatabaseServiceMib* group has objects for the database service considering network general information.

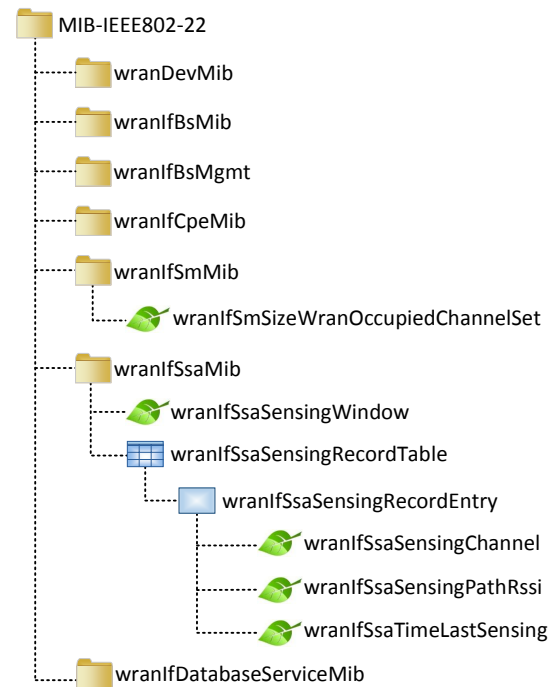


Fig. 1. IEEE 802.22 MIB groups structure

Considering the IEEE 802.22 MIB module, we can highlight some important objects for a CR network management system. These objects provide useful information regarding the spectrum sensing in each CR device. An important object is the occupied channel list (*wranIfSmWranOccupiedChannelSet*), which identifies all channels evaluated as occupied in the spectrum sensing result. The object associated with the sensing window configuration (*wranIfSsaSensingWindow*) determines how long each channel must be sensed to obtain the evaluation

result. The sensed channel number (*wranIfSsaSensingChannel*), the Received Signal Strength Indication (RSSI) during the sensing (*wranIfSsaSensingPathRssi*), and the timestamp of most recent sensing (*wranIfSsaTimeLastSensing*) are also fundamental information to develop a CR network management system.

The communication between the Agent and the Gateway is realized through the Simple Network Management Protocol (SNMP). This protocol is particularly interesting to be used among BS and CPEs entities because all the messages exchanged among them are usually transmitted over a dedicated control channel [4], avoiding overhead of control messages in the communication. This control channel is selected before the initialization of the network operation and presents the best propagation conditions. Usually, this channel comprises a 6 MHz bandwidth channel, with transfer rate of 18Mbps and can be used only for control and management data between BSs and CPEs. Moreover, considering an IEEE 802.22 network with a maximum of 512 Agents and 2 seconds as the smallest time interval between two consecutive requests performed by the Gateway, the SNMP traffic will correspond to less than 5% of the control channel capacity. However, SNMP is not suitable to realize communications over the Internet, mainly due to the absence of a control channel and also because firewalls may block SNMP messages. In addition, SNMP security mechanisms causes network overhead [15].

One way to avoid SNMP over the Internet is using management approaches based on Web Services (WS) [16]. These approaches may be implemented through two architectures: Service Oriented Architecture (SOA) [17] and Resource Oriented Architecture (ROA) [18]. Both architectures may use Hypertext Transfer Protocol (HTTP) to transmit data, avoiding firewalls. Moreover, SOA and ROA take advantage of HTTP Secure (HTTPS) protocol to easily secure the messages being exchanged. However, Pautasso *et al.* [19] proved that ROA is better than SOA to achieve a higher performance when implementing a WS based approach.

ROA is a loosely coupled approach to the client-server model that uses a Uniform Resource Identifier (URI) to directly access devices resources. In general, this approach follows the Representational State Transfer (REST) architectural style [18]. This style defines HTTP as the only application protocol and standardizes the access interface through methods of this protocol (*i.e.*, GET, PUT, POST, and DELETE). Each REST message represents a state of the accessed resource, *i.e.*, the current collection of meaningful information (*e.g.* sensed channel list, sensing window, and sensing evaluation). Other features of REST includes the support to cache messages and resources inheritance. Despite all advantages, ROA was not used, at our best knowledge, to implement a sophisticated management application to CR networks. Therefore, we investigate and explain the related work about CR networks management in the next subsection.

## B. Related Work

Much has been discussed about CR in the last decade. The architecture proposed by Wang *et al.* [9] uses the information provided by the CR devices to automatically configure the network access. The architecture was developed on a multi-processor System-on-Chip but the authors highlight the need of a CR network management system. Potier and Quian [8] addressed the problem of management for CR ad-hoc networks and presented how the management of these networks is different than the traditional management for wired and wireless networks. Moreover, the authors proposed a cognitive network management protocol for ad-hoc networks. Wang *et al.* [10] provided an overview of the essential functionalities of the spectrum sensing and discussed its impact on making business level policies for CR networks management. Manfrin, Zanella, and Zorzi [11] proposed CRABSS<sup>1</sup>, an open platform to monitor unlicensed spectrum bands. The proposed system export information about the spectrum sensing function to management tools, facilitating the development of spectrum utilization maps. Stavroulaki *et al.* [2] presented an overview about machine learning mechanisms to perform automatic decisions about configuration in CR networks. The authors also presented basic learning functionalities for the identification and processing of information that can lead to exploitable knowledge in CR networks.

The researches described above are specific solutions to CR management, *e.g.* the automatic configuration to access the CR network, an ad-hoc protocol, the application of policies and machine learning in these networks. In a CR network, CR devices should operate automatically, but we argue that this operation can be improved by combining the analysis of spectrum sensing results and the administrator intervention. An administrator may configure the cognitive functions parameters such as the range of channels to be sensed, sensing window, list of sensing channels, etc. However, to the best of our knowledge, no management system for CR networks based on the spectrum sensing function has been proposed. Therefore, there is a lack of management systems that enable the configuration, monitoring, and visualization of CR networks. To minimize this lack, we propose a management system called *Kitsune*, which allows administrators to acquire a better knowledge about a managed CR network. In the next section, we present the *Kitsune* system, the modules of its architecture, and the prototype developed and deployed in a real CR network.

## III. KITSUNE MANAGEMENT SYSTEM

In this section, *Kitsune* management system is described in details. First, in the Subsection III-A we present and discuss the functional blocks of *Kitsune*, detailing each component of the system. Next, we describe the details related to the prototype developed to validate the operation of the proposed system in the Subsection III-B.

<sup>1</sup>ARAGORN Proejct – <http://www.ict-aragorn.eu/>

## A. Kitsune Functional Blocks

*Kitsune* system aims to manage CR networks based on spectrum sensing. This system follows a hierarchical management model composed of one functional block for the Network Operation Center (NOC) and another block representing the CR network. These two functional blocks exchange information through a backhaul, which may be a private network infrastructure, *e.g.* a cellular operator network, or a shared infrastructure, such as the Internet. Fig. 2 illustrates in details *Kitsune* proposed system using a top-down approach.

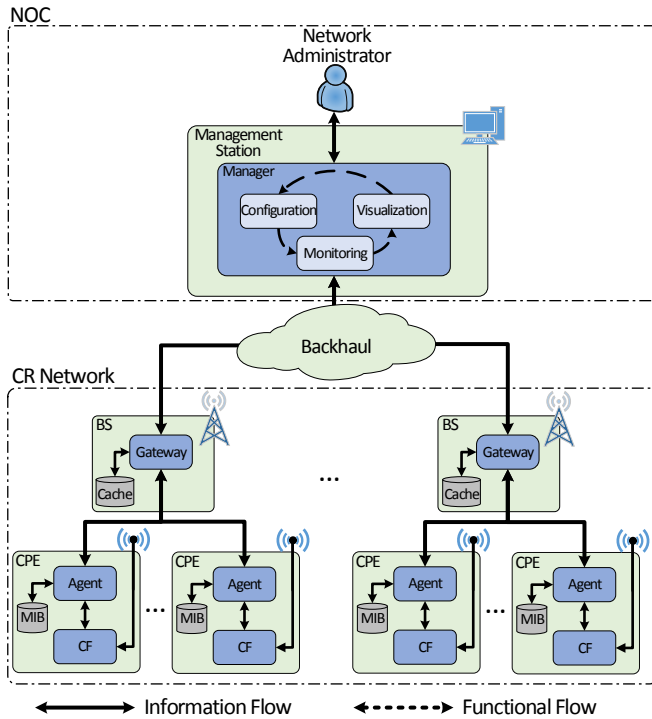


Fig. 2. *Kitsune* management system

The NOC comprises a physical entity called Management Station, as shown in Fig. 2. This Management Station presents a Manager component composed of three modules, Configuration, Monitoring, and Visualization. Through these three modules, a network administrator can improve the configuration of the spectrum sensing function of each CR device by monitoring and analyzing its results. Therefore, the functional flow inside the Manager represents a continuous learning process to the network administrator.

In the first stage of this learning process, the network administrator defines the configurations, using the Configuration module. In this sense, the network administrator defines which parameters of the spectrum sensing function will be managed (*e.g.* sensed channel list, sensing window, and sensing evaluation). In the second stage of the learning process, the Monitoring module will monitor the spectrum sensing results in each device of the CR network. Finally, the third stage is provided by the Visualization module which will show a composition of the historical information about the spectrum

sensing results of each CR device. Since the learning process exhibits a cyclic behavior, the Visualization may be used as a feedback by the network administrator to perform new configurations on the CR network.

As defined by IEEE 802.22 standard, a centralized CR network is composed by two physical entities: BS and CPE [4]. BS is characterized by concentrating all the network information in an infrastructural wireless communication system. In this BS, we design the second component of *Kitsune*, called Gateway. This component is responsible for four main tasks: (i) delivering the configuration messages sent from the Manager to the spectrum sensing of each CPE, (ii) requesting the spectrum sensing results of each CPE, (iii) caching these results, and (iv) forwarding the results to the Manager component in a predefined time interval. In addition, the caching of the spectrum sensing results is an important task in the hierarchical management model, because it enables the Manager to obtain results from the Gateway without major overheads caused by messages exchanged directly with CPEs.

CPE is a physical entity responsible for providing network access to users. This entity is composed of three components: (i) MIB, (ii) Agent and (iii) Cognitive Functions (CF). A MIB is a conceptual database that organizes the information about the current settings and results of the spectrum sensing function. The Agent component receives configuration messages from the Gateway and organizes them into the MIB. Moreover, the Agent collects spectrum sensing results from the CF component and also organizes these results in the MIB. Finally, the CF component performs the spectrum sensing function through the use of SDR. In the next subsection, we describe the prototype implementation of the *Kitsune* system.

## B. Prototype Implementation

*Kitsune* prototype implements both the NOC and the CR network functional blocks, as previously defined. The information exchange among system components may be performed through internal communication (*i.e.* inside the same device or computer) or external communication (*i.e.* among different devices and computers), as shown in Fig. 3.

The Manager summarizes the spectrum sensing results and creates a visualization for the network administrator through a Web based Graphic User Interface (GUI). These summarizations are stored in a MySQL database and queried by the Manager in a predefined time interval. The Manager and Gateway communications follows ROA because it enables a better usage of the Internet infrastructure. In addition, Managers and Gateways communication is performed over an encrypted connection using HTTPS. Finally, communication among Gateways and CPEs is typically performed over LANs, making SNMP preferable in this case. As described previously, the Gateway possesses four tasks, which are implemented through a MySQL database in combination with services provided to the Manager. In addition, services and parameters supported by a Gateway are presented in Table I.

Accordingly to Table I, when a Manager requests the service *beginSession*, the Gateway creates a session, authenticated by

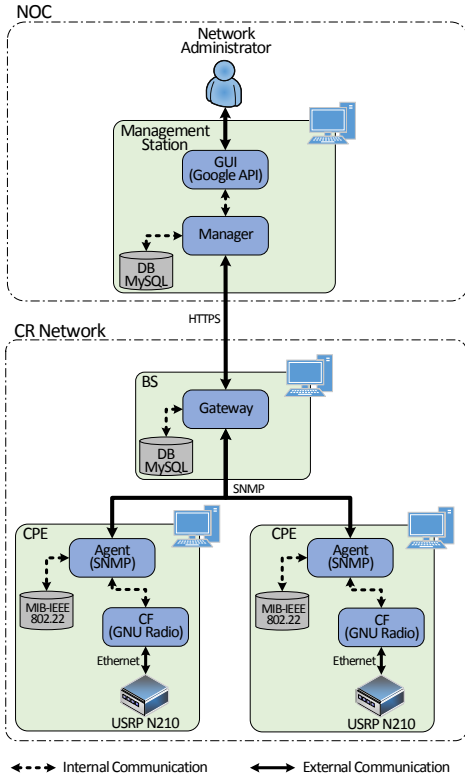


Fig. 3. *Kitsune* system prototype

TABLE I  
GATEWAY SERVICES

Service	Parameter	Return
<i>beginSession</i>	<i>username, password</i>	<i>sessionID</i>
<i>endSession</i>	<i>sessionID</i>	-
<i>setConfiguration</i>	<i>sessionID, configuration</i>	<i>configStatus</i>
<i>setCpeConfiguration</i>	<i>sessionID, cpeID, cpeConfiguration</i>	<i>cpeConfigStatus</i>
<i>setCpeList</i>	<i>sessionID, newCpe</i>	<i>cpeList</i>
<i>getCpeList</i>	<i>sessionID</i>	<i>cpeList</i>
<i>getCpe</i>	<i>sessionID, cpeID</i>	<i>cpeSensingInfo</i>
<i>getSensingInfo</i>	<i>sessionID</i>	<i>sensingInfo</i>

the *username* and *password* parameters. This service starts an authorized session that can be finished using the *endSession* service. To configure the Gateway, the *setConfiguration* service can be used. Moreover, the *setCpeConfiguration* service can be performed to directly configure each CPE. The *setCpeList* and *getCpeList* services are implemented to directly access the list of Agents presented in the Gateway. The *getCpe* service returns the spectrum sensing results about a CPE, identified by the *cpeID* parameter. The results of this service may be an occupied channel list (*wranIfSmWranOccupiedChannelSet*), the sensing window configuration (*wranIfSsaSensingWindow*), the received signal intensity (*wranIfSsaSensingPathRssi*), among others objects of the IEEE 802.22 MIB. The MIB presented in each Agent is based on the IEEE 802.22 MIB, as shown in Fig 1. Finally, the *getSensingInfo* service returns the spectrum sensing results about all the CPEs monitored in the Gateway.

The Agents are deployed in Ettus USRP N210 devices

with a computer-hosted SDR kit that enable the fast development of radio applications [20] through an SDR framework called GNUradio [21]. In addition, these devices perform the spectrum sensing function. The USRP device operates according to the algorithms implemented in the personal computer connected with the USRP through a Gigabit Ethernet interface. We developed a Python script to perform the spectrum sensing function that interacts with the MIB. This script has implemented an energy detector based spectrum sensing technique, given its wide adoption [3]. The script uses the parameters configured by the network administrator, such as the list of channels to sense, sensing period, and sensing window. Finally, results of the SSF are stored in the MIB and both the configurations and results are manipulated by the Agents.

Once defined the *Kitsune* system and the prototype implementation, in the next section we present in details how we deployed *Kitsune* system prototype in three different scenarios to be evaluated. In addition, we describe the experiments performed over each scenario where *Kitsune* was evaluated and discussed in details according to the obtained results.

#### IV. KITSUNE EVALUATION

*Kitsune* prototype was deployed and evaluated according to three scenarios. Each scenario was based on the IEEE 802.22 standard, where analogic TV channels are used to provide Internet access [4]. In Table II, parameters and their respect values are representing hardware and software configurations used for these three scenarios during all experiments.

TABLE II  
KITSUNE EVALUATION SCENARIO PARAMETERS AND VALUES

Parameter	Value
Execution time	60s
USRP sample rate	$10^6$ samples/s
Number of sensed channels	5
$\lambda$	[1, 2, 3, 4, 5]s
Sensing Window	.11s
Sensing Period	[0.1, 2]s
$P_M$	30s
$P_G$	2s

According to Table II, for each scenario experiments were performed during 60 seconds, considering a USRP sample rate of  $10^6$  samples/second. We defined 5 different channels to be sensed by 5 CPEs. In addition, users behavior in the sensed channels was modeled through a Poisson distribution [22] that changes the vacancy status of the channels with a mean and variance ( $\lambda$ ) for each channel. These channels were sensed during a sensing window of 0.1 second and the interval between each spectrum sensing function execution was set to be a different period for each scenario. Another two important parameters are the polling waiting time for Managers requests to Gateway ( $P_M$ ), setted to 30 seconds, and the polling waiting time for Gateway requests to Agents ( $P_G$ ), setted to 2 seconds. We defined the  $P_M$  higher than the

$P_G$  on purpose to verify that the Gateway summarizes properly the informations obtained in a smaller interval.

In the Subsection IV-A, the first experimental scenario used to visualize the channel evaluation is described and results obtained are discussed. Similarly, in the Subsection IV-B we described the second scenario and discussed the results gathered during the evaluation of transmissions performed by the CPEs. Finally, in Subsection IV-C we evaluate how the network performance can be improved by a network administrator that analyzes the visualizations and decide new configurations for the spectrum sensing function.

### A. Channel Status Evaluation

In this scenario, the BS defines a silent phase where CPEs are not able to perform transmission. In this phase, CPEs should periodically perform the spectrum sensing over the channels defined by the BS, using a sensing period setted to 2 seconds. The spectrum sensing results are requested from CPEs by a BS to determine the actual status of the channels occupancy. The main goal of this scenario is to verify that *Kitsune* allows the network administrator to analyze results of the spectrum sensing performed by CPEs, enabling the improvement of the knowledge about users behavior in the sensed channels. Fig. 4 shows three BS visualizations through the *Kitsune* interface.

In Fig. 4, we can observe three visualizations provided by *Kitsune* interface from the BS perspective. The occupancy status of the sensed channels, during a time period of 60 seconds, can be observed in Fig 4(a). Through this visualization, a network administrator may observe the user behavior in the sensed channels and make some considerations. For example, the administrator may conclude that channel 4 is a bad option to be selected for transmission, because it has longer periods of occupancy when compared to others. It means that a CPE in channel 4 would have to often stop its transmission due to the heavy usage of the channel, decreasing its overall throughput. Different from channel 4, channel 2 presents a lower usage, becoming a potential option to be selected for transmissions.

*Kitsune* also provides a visualization of the RSSI in each sensed channel. This is an important information that can be used to analyze the quality of the sensed channels, as can be seen in Fig. 4(b). Associating this visualization with Fig. 4(a) it is possible to stipulate the sensibility of the spectrum sensing, *i.e.*, the minimum RSSI necessary to consider a channel as occupied. The network administrator may use this information to adjust the sensing sensibility to a desired level. In addition, the administrator may highlight the information of a specific channel to analyze, as exemplified in Fig. 4(b), where the highlighted line represents the RSSI of channel 5.

Associating Fig. 4(a) and Fig. 4(b) we can observe the spectrum sensing defining channel 5 as occupied, when the sensed channel presents a high RSSI (*e.g.* about 16 dBm). On the other hand, when a low RSSI is observed in the channel (*e.g.* about 6 dBm), it will be defined as vacant. It is a characteristic of the energy detector spectrum sensing technique used in this paper, which compares the energy



Fig. 4. BS visualizations

sensed in the channel with a threshold to define the channel status as vacant or occupied.

The geolocation of BS and CPEs may be used by *Kitsune* to display a map of the network, as can be seen in Fig. 4(c), based in an interpolation method of geolocation information associated to Okumura-Hata propagation model to define coverage area. In this map, the BS is represented by a tower icon and its coverage area is delimited by the colored circle. Its associated CPEs are represented by house icons. This visualization can be combined with the channel vacancy status to provide the network administrator an overview about the network. The combined use of these visualizations is a powerful tool to achieve a better understanding about the sensed channels.

### B. CPE transmissions

The second scenario is based on the transmission performed by each CPEs that compose the CR network. In this case, CPEs

should perform the spectrum sensing to verify the status of the channel and transmit only if the channels status are defined as vacant. Once verified the status of the channel, the CPE starts its transmission over it until a new spectrum sensing is performed. The interval between each spectrum sensing is given by the sensing period. In this scenario, *Kitsune* shows the information about CPEs transmissions over the channel configured for each CPE. Moreover, with the analysis of the transmissions, administrators can reconfigure the network to maximize the throughput obtained by a specific CPE. Results of the transmission analysis are shown in Fig. 5.

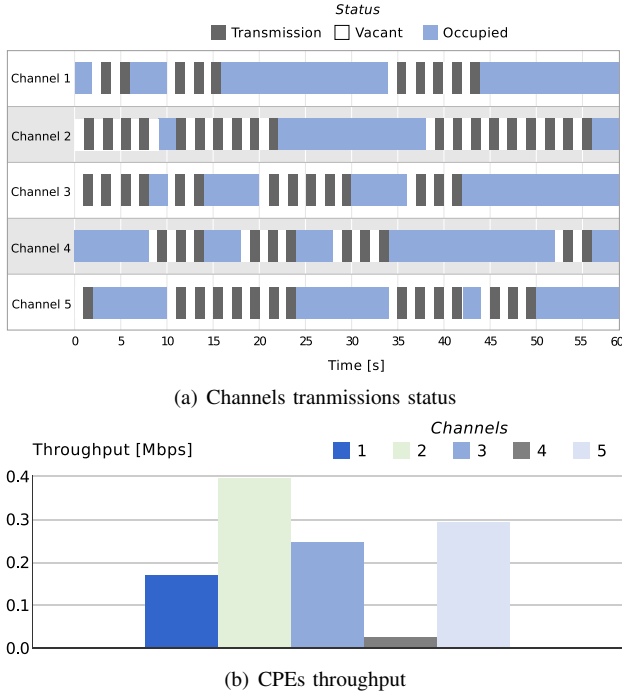


Fig. 5. CPE transmissions visualization

As can be seen in the Fig. 5, we designed two different visualizations of the transmissions performed by CPEs.. Fig. 5(a) shows a visualization similar to Fig. 4(a). However, in addition to showing the results of spectral sensing, this new visualization enable an important analysis not available through Fig. 4(a). This analysis is related to transmissions performed by CPEs over the channels, highlighted with a different color for vacant and occupied status. With an analysis over this visualization, the network administrator can determine which is the channel that presents fewer opportunities for transmission. Once the network administrator knows which is the worst channel, in terms of number of transmissions, this channel can be avoided, and another channel can be defined to be analyzed by a CPE.

Complementary to the visualization shown in Fig. 5(a), Fig. 5(b) allows another important analysis about the transmissions performed by the CPEs. This information is related to the total throughput obtained in the transmissions. We can associate this throughput directly to the transmissions shown in Fig. 5(a), identifying which is the channel with higher throughput. On

the other hand, the worst channel, in terms of throughput and number of transmission can be identified by the administrator that may configure the network to ignore this channel, avoiding wasting time by sensing a bad channel for transmission. In the example of Fig. 5, we can observe that the channel 2 presents a higher throughput than the others, being a better choice to be used for transmissions. However, channel 4 presents the lower total throughput. The administrator can analyze the occupancy, throughput, and the RSSI of channels in the visualization and define which is the best channel. For example, A channel with 95% of occupancy, 6dB of RSSI and 30Kbps is a bad channel while a channel with 15% of occupancy, 18dB of RSSI and 1Mbps is a good channel.

### C. Configuration Analysis

Finally, the last scenario is also based on the transmissions performed by CPEs. However, in this evaluation we are interested in analyzing the impact of a new configuration settled by the network administrator after analyzing the initial configuration. To perform this evaluation, we stored the results of the spectrum sensing function to apply the same channel occupancy for both configurations. Then, we setted an initial sensing period and obtained the average throughput by a CPE in each channel. Results were gathered and visualized through *Kitsune* system. Afterwards, a new configuration for the sensing period was set to monitor the results. These results are shown in Table III.

TABLE III  
THROUGHPUT RESULTS WITH DIFFERENT CONFIGURATIONS

Channel	Sensing Period [s]	Throughput [Mbps]	Variation [%]	Transmissions
1	1	0.3182	42.04	18
	2	0.5490		25
2	1	0.2267	51.84	15
	2	0.4708		24
3	1	0.4016	18.42	22
	2	0.4923		24
4	1	0.1803	42.54	12
	2	0.3138		17
5	1	0.4027	17.25	21
	2	0.4867		23

According to Table III, we defined two different configurations for the sensing period: 1 second and 2 seconds. In both cases, the throughput obtained for each channel varied. This variation is given by the behavior of the network users, which directly impacts in the results. Moreover, one important conclusion from the analysis of the generated visualizations: the sensing period parameter of a CPE impacts in the average throughput obtained during the transmissions. In the fourth column of the Table III is shown the variation of the average throughput obtained in each sensing period. For all the channels, the average throughput increases when the sensing period is higher. This can be explained due the fact that a CPE will interrupt their transmissions to perform the sense with less frequency, allowing a higher transmission time.

Another important observation that can be highlighted is that the variations between channels varies for each channel. For example, the difference of the average throughput obtained

in the channel 1 was 42.04%, while the channel 3 presents a variation of 18.42%. To understand these variations, we should observe the number of transmissions performed in each channel. These numbers varies according to the behavior of the users on the channel. For example, in channel 1, with a sensing period of 1 second, 18 transmissions were performed. Meanwhile, 22 transmissions were performed in channel 3, with the same sensing period. Increasing this period to 2 seconds, increased by 7 the number of transmissions of the former channel, meanwhile it increased only by 2 the number transmissions of channel 4. Thus, the number of transmission directly impacts in the average throughput of the CPEs. Therefore, *Kitsune* system is able to provide an overview about throughput variations, their cause and consequence, which combined with former scenarios results from subsection IV-A and IV-B become powerful tools for the administrator to understand the behavior of the users in a CR network and to participate in all three stages of the continuous learning process.

Once proved the proper operation of the *Kitsune* system, in the next section we present the final conclusions obtained through the development of this work and discuss future work related to the improvement of *Kitsune*.

## V. CONCLUSION AND FUTURE WORK

In this paper, we proposed *Kitsune*, a management system for CR networks that helps administrators to learn how to improve the configuration of the spectrum sensing function. *Kitsune* allows administrators to participate in a continuous learning process based on the configuration, monitoring and visualization of the spectrum sensing function through a Graphical User Interface. A prototype of *Kitsune* was implemented and evaluated through a real RF environment based on IEEE 802.22 WRAN scenario, showing the operation of the proposed management system.

Results show that *Kitsune* system allows network administrators to improve the throughput obtained in transmissions performed by CPEs by gathering results (monitoring), applying visualizations techniques (visualization), and finally, changing the configurations of these devices (configuration). In summary, *Kitsune* enables administrators to participate actively in the spectrum sensing monitoring, controlling, and visualization. Such a participation allows *Kitsune* to readjust a CR network to achieve a better throughput, in average, as described in Section IV.

There are several avenues for future research. We intend to further extend *Kitsune* operation, allowing the configuration, monitoring and visualization of the rest of cognitive functions performed by cognitive devices. Moreover, *Kitsune* can be improved in terms of functionalities through new mechanisms that improve the network operation. For example, machine learning algorithms may be implemented in *Kitsune* to identify the best vacant channels in terms of historical availability, propagation conditions, and also select channels to be used by CPEs according to Quality of Service (QoS) policies.

## ACKNOWLEDGEMENTS

The authors thanks to Digitel S.A., Fundação de Apoio a Pesquisa do Estado do Rio Grande do Sul (FAPERGS), Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) and Coordenação de Aperfeiçoamento de Nível Superior (CAPES) for the financial support.

## REFERENCES

- [1] I. Mitola, J. and J. Maguire, G.Q., "Cognitive radio: making software radios more personal," *IEEE Personal Communications*, vol. 6, pp. 13–18, 1999.
- [2] V. Stavroulaki, A. Bantouna, Y. Kritikou, K. Tsagkaris, P. Demestichas, P. Blasco, F. Bader, M. Dohler, D. Denkovski, V. Atanasovski, L. Gavrilovska, and K. Moessner, "Knowledge Management Toolbox: Machine Learning for Cognitive Radio Networks," *IEEE Vehicular Technology Magazine*, pp. 91–99, June 2012.
- [3] T. Yucek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," *IEEE Communications Surveys & Tutorials*, vol. 11, pp. 116–130, 2009.
- [4] IEEE, "Standard for Information Technology - Telecommunications and information exchange between systems Wireless Regional Area Networks (WRAN) - Specific requirements Part 22," *IEEE Std 802.22*, pp. 1–680, 2011.
- [5] M. T. Masonta, M. Mzyece, and N. Ntlatlapa, "Spectrum decision in cognitive radio networks: A survey," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1088–1107, 2013.
- [6] M. Dardaillon, K. Marquet, T. Risset, and A. Scherrer, "Software defined radio architecture survey for cognitive testbeds," *International Wireless Communications and Mobile Computing Conference (IWCMC)*, pp. 189–194, 2012.
- [7] J. Wang, M. Ghosh, and K. Challapali, "Emerging cognitive radio applications: A survey," *IEEE Communications Magazine*, pp. 74–81, March 2011.
- [8] P. Potier and L. Qian, "Network management of cognitive radio ad hoc networks," *International Conference on Cognitive Radio and Advanced Spectrum Management (CogART)*, pp. 1–5, 2011.
- [9] S. Wang, L. Xie, H. Liu, B. Zhang, and H. Zhao, "Acra: An automatic and expandable architecture for cognitive radio nodes," *International Conference on Wireless Communications and Signal Processing (WCSP)*, pp. 1–5, October 2010.
- [10] C.-X. Wang, H.-H. Chen, X. Hong, and M. Guizani, "Cognitive radio network management," *IEEE Vehicular Technology Magazine*, vol. 3, no. 1, pp. 28–35, 2008.
- [11] R. Manfrin, A. Zanella, and M. Zorzi, "CRABSS: CalRadio-Based advanced Spectrum Scanner for cognitive networks," *Wireless Communication & Mobile Computing*, vol. 10, no. 12, pp. 1682–1695, 2010.
- [12] I. Akyildiz, W.-Y. Lee, M. Vuran, and S. Mohanty, "A survey on spectrum management in cognitive radio networks," *IEEE Communications Magazine*, pp. 40–48, 2008.
- [13] J. Peha, "Sharing spectrum through spectrum policy reform and cognitive radio," *Proceedings of the IEEE*, vol. 97, no. 4, pp. 708–719, 2009.
- [14] I. Christian, S. Moh, I. Chung, and J. Lee, "Spectrum mobility in cognitive radio networks," *IEEE Communications Magazine*, vol. 50, no. 6, pp. 114–121, 2012.
- [15] G. Pavlou, P. Flegkas, S. Gouveris, and A. Liotta, "On management technologies and the potential of web services," *IEEE Communications Magazine*, vol. 42, no. 7, pp. 58–66, 2004.
- [16] G. Moura, G. Silvestrin, R. Sanchez, L. Gaspary, and L. Granville, "On the Performance of Web Services Management Standards - An Evaluation of MUWS and WS-Management for Network Management," *IFIP/IEEE International Symposium on Integrated Network Management (IM)*, pp. 459–468, 2007.
- [17] D. Box, D. Ehnebuske, G. Kakivaya, A. Layman, N. Mendelsohn, H. Nielsen, S. Thatte, and D. Winer, "Simple object access protocol (SOAP) 1.1," 2000.
- [18] R. Fielding, "Architectural styles and the design of network-based software architectures," Ph.D. dissertation, University of California, 2000.

- [19] C. Pautasso, O. Zimmermann, and F. Leymann, "Restful web services vs. "big" web services: making the right architectural decision," *Proceedings of the 17th international conference on World Wide Web*, pp. 805–814, 2008.
- [20] Ettus Research, "USRP2," <http://www.ettus.com>.
- [21] Free Software Foundation, "GNU radio," [www.gnuradio.org](http://www.gnuradio.org), 2004.
- [22] C. Ghosh, S. Pagadarai, D. Agrawal, and A. Wyglinski, "A framework for statistical wireless spectrum occupancy modeling," *IEEE Transactions on Wireless Communications*, vol. 9, pp. 38–44, 2010.