Efficient Mid-end Spectrum Sensing Implementation for Cognitive Radio Applications based on USRP2 Devices

Daniel Denkovski, Vladimir Atanasovski and Liljana Gavrilovska
Faculty of Electrical Engineering and Information Technologies
Ss Cyril and Methodius University in Skopje
Skopje, Macedonia
{danield, vladimir, liljana}@feit.ukim.edu.mk

***Abstract*—Spectrum sensing is a cornerstone task in cognitive radio environments supporting dynamic spectrum access by spectrum opportunities discovery. It must be reliable and accurate in order not to harm the primary system by incorrect spectrum opportunities decisions. Mid-end spectrum sensing devices are spectrum sensing devices with satisfactory signal detection performances and reasonable price. This paper presents an efficient and flexible mid-end spectrum sensing solution (an USRP2 based spectrum sniffer implementation) that offers many sensing functionalities and detection capabilities by implementing several sensing modes of operation and detector types. Additionally, the paper presents several usage possibilities of this USRP2 spectrum sniffer implementation proving its efficient employment in various spectrum sensing applications.**

***Keywords- spectrum sensing, cognitive radio, USRP2 based sniffer implementation, detector, operation mode*.**

I. INTRODUCTION

Dynamic spectrum access is an increasingly popular networking paradigm aiming to resolve frequency spectrum scarcity problems that come from the traditional fixed frequency resources reservation policies. Cognitive radios incorporate the dynamic spectrum access functionality striving to access and use the temporary available frequency bands and dynamically adapt to changing environment conditions.

Spectrum sensing is a crucial aspect in cognitive radio since it enables the secondary users to be spectrum aware and facilitates the spectrum opportunities discovery process. There are various and numerous spectrum sensing implementations that can be generally classified into three groups: high, low and mid-end sensing devices. High-end spectrum sensing devices mainly refer to spectrum analyzers and these devices' main characteristic is their high precision in terms of signal detection. However, the high accuracy reflects in high price, which may make them inappropriate for certain cognitive radio applications. On the other hand, the low-end spectrum sensing devices are cheap solutions having low precision and, thus, limited applicability in different applications. They often result in false conclusions about the spectrum opportunity. The mid-end spectrum sensing devices are a trade-off between price and accuracy. They are cheaper sensing solutions than the high-end spectrum analyzers and offer solid detection performances. All spectrum sensing solutions may employ various spectrum sensing techniques [1]. However, energy detection is most commonly used technique due to its simplicity and satisfactory detection capabilities.

The spectrum sensing devices can be used in many spectrum related applications. Besides their main function to detect spectrum opportunities in cognitive radios, they can be also used in long and medium term spectrum measurements [2, 3]. Furthermore, different cooperative sensing [4, 5] and data fusion techniques can be implemented to achieve better detection capabilities in the cognitive environment. The input from the sensing devices can be also used for planning of medium and long term spectrum sharing strategies, either between primary and secondary users or between the secondary users in the cognitive network.

This paper presents an efficient and flexible mid-end USRP2 based spectrum sensing (i.e., sniffer) implementation. The developed sniffer implementation extends the spectrum sensing capabilities of the basic GNU radio USRP2 sensing implementation (used in [6]) introducing several detector types and different modes of operation of the sniffer. This sensing implementation can be used to perform wideband and band-specific spectrum measurements and can be utilized in various cognitive radio applications.

The paper is organized as follows. Section II presents the spectrum sniffer implementation based on USRP2 hardware with all comprising GNU radio based and extended processing blocks and the functionalities and possibilities of the sniffer. Section III presents several possible sensing relevant applications of the USRP2 based sniffer implementation. Section IV concludes the paper and presents directions for the future work.

II. USRP2 BASED SPECTRUM SNIFFER IMPLEMENTATION

This section elaborates on a custom design and development of a versatile USRP2 based spectrum sensing (i.e., sniffer) implementation using the off-the-shelve available USRP2 hardware and accompanying GNU Radio software.

 ISBN: 978-1-61208-131-1

Universal Software Radio Peripheral 2 (USRP2) [7] is a hardware platform for development of various radio applications. Its motherboards are enabled to use various daughterboards covering different frequency ranges. The USRP2 hardware supports fast and high precision analog-to-digital and digital-to-analog conversion and Ethernet interface for connection to host computers. This results in wider band signals support, as well as increased dynamic range of the system. GNU Radio [8] is an open source software development toolkit that provides the basic signal processing blocks required to make the USRP2 hardware a software defined radio (SDR) programmable platform. GNU Radio applications are primarily written in python with the performance-critical signal processing written in C++.

The USRP2 based sniffer implementation is developed as a C++ application that includes the GNU Radio basic and USRP2 specific blocks. This sensing implementation is based on the logic of the original GNU radio python based implementation *usrp_spectrum_sense.py*, with extended functionalities and support of several additional features. Moreover, the C++ realization of the sniffer eliminates the C++ to/from python parsing resulting in better sensing performances. This subsection presents the sniffer architecture and the possibilities provided by the sniffer, i.e., the supported detector types and modes of operation.

1) USRP2 based sniffer architecture

The sensing-relevant input parameters in the developed USRP2 C++ based sniffer implementation are: *starting* and *end frequency*, *decimation factor*, *FFT size*, *gain*, *detection type*, *tune time* (if switching between frequencies is required) and *dwell time* (time spent at a measurement frequency point). More details on the detection types of the sniffer and the modes of operation are given in subsections 2) and 3), respectively.

The developed USRP2 C++ sniffer implementation architecture along with all the included GNU Radio basic and extended blocks is depicted on Figure 1. The architecture consists of seven processing blocks: usrp2_source_32fc, gr_stream_to_vector, gr_fft_vcc, gr_fft_vfc, gr_complex_to_mag_squared, gr_complex_to_mag and the custom made gr_energy_detector_f processing block. Their key characteristics are:

- *usrp2_source_32fc* is a GNU radio USRP2 specific block that creates the USRP2 source and has the control of the hardware (it sets up the decimation rate and performs the tuning of the center frequency).
- *gr_stream_to_vector* is a GNU radio block that converts the input from the usrp2_source_32fc from stream of complex samples to vector of complex samples with size fft_size.
- *gr_fft_vcc* is a GNU radio processing block that transforms the vector of time samples with size fft_size into frequency domain vector of complex samples with size fft_size.
- *gr_complex_to_mag* is a GNU radio processing block that converts the complex vector of time samples with size fft_size into amplitudes vector with the same size.
- *gr_fft_vfc* is a GNU radio block that converts the vector of amplitudes per time to frequency domain complex vector with size fft_size.
- *gr_complex_to_mag_squared* is a GNU radio processing block that converts the complex vector (sample) in power vector (sample) summing the magnitude squared I and Q values and passing the output to gr_energy_detector_f block.
- *gr_energy_detector_f* is a custom made C++ processing block that performs the actual preprocessing of the measurement data. On initialization of this block the detection type, dwell and tune times are specified. This block constantly receives the vectors' samples from the gr_complex_to_mag_squared block and performs the chosen preprocessing (detection type) during the specified dwell_time. The output of this block is packed into a message queue variable of type gr_msg_queue, timely read and post-processed by the main function. The output of the main function is the decisions vector for the occupancy in the inspected frequency span. The gr_energy_detector_f also initiates the tuning between sequential frequencies if required.

Figure 1 illustrates the possibilities and features that the current USRP2 C++ sniffer implementation can offer. Several flow graphs are possible, based on the selection of the detection type and the mode of operation of the sniffer (subsections 2) and 3)). The current implementation of the sniffer also provides possibilities for *remote control*.

2) Detector types

The current USRP2 based sniffer implementation allows selection of *five energy-based detector types*. The main enabler of the different detection types is the custom made *gr_energy_detector_f* block. Besides the maximum hold detector as the basic implementation in GNU radio, it enables the following four other detection types to be used: minimum hold, mean hold, Higher-Order-Statistics Energy Detection (HOSED) type, as well as FFT Averaging Ratio (FAR) detection type.

The *minimum hold detector* saves and returns the minimum value of the received power during the inspection time (dwell time) on a specific frequency band. Oppositely, the *maximum hold detector* returns the maximal value of the received power during the examination time. The *mean hold detector* calculates and gives at the output the average received power during the inspection time. When the previous three detection types are in use, the occupancy decision on frequency band of interest is based on comparisons with predefined threshold values. The *higher order statistics energy detector*, besides the average received power, calculates important higher order statistics such as *skewness* and *kurtosis* [9] of the received power during the inspection time. This valuable statistical

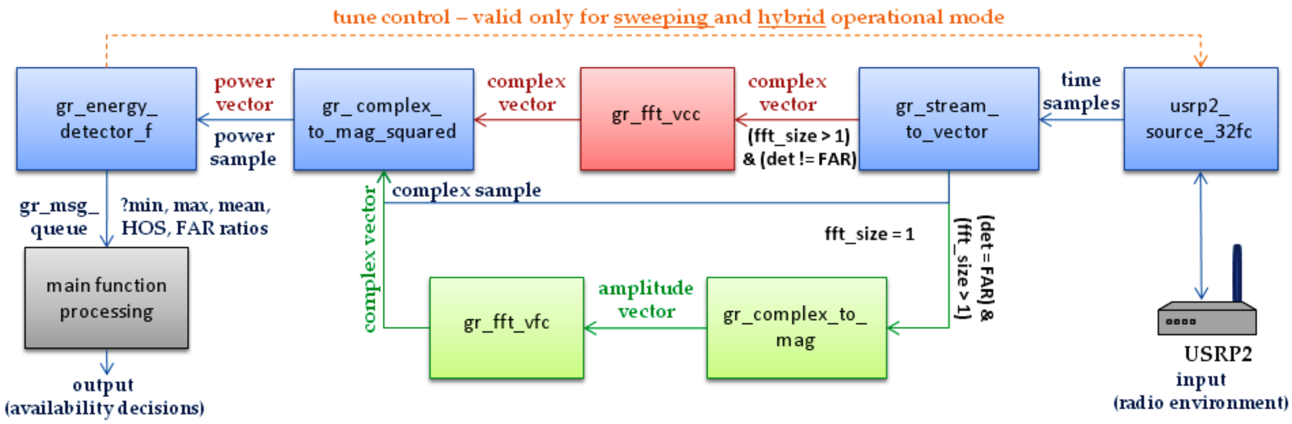

Figure 1. USRP2 based sniffer application architecture

information can be helpful in the detection process since it reflects some characteristics of the distribution of the receiving power and has proved to offer better detection performance than the classical average energy detector. The decision on the occupancy of the inspected frequency band in the case of HOS detection is made by comparison of the calculated statistics values with the statistics values of the system noise. The *FFT averaging ratio detector* [10] is a detector utilizing FFT analysis on the amplitude of the received time samples. The output of this energy detector type is the average PSD (Power Spectral Density) ratio of each frequency bin in the FFT analysis. Based on these output values, a decision is made whether a frequency band is occupied or not by comparing the referred with predefined threshold values.

When using the first four detection types, the flow graph of the sniffer is consisted of the following five blocks: *usrp2_source_32fc*, *gr_stream_to_vector*, *gr_fft_vcc*, *gr_complex_to_mag_squared* and *gr_energy_detector_f* block, if using FFT. If FFT is not used (fft_size=1), the *gr_fft_vcc* is excluded from the previous flow graph (Figure 1). When FAR detection is selected the sniffer flow graph consists of the following six blocks: *usrp2_source_32fc*, *gr_stream_to_vector*, *gr_complex_to_mag*, *gr_fft_vfc*, *gr_complex_to_mag_squared* and *gr_energy_detector_f* blocks, as the FFT analysis is performed on the amplitude of the time samples (Figure 1).

3) Modes of operation

The custom made C++ sniffing implementation enables the USRP2 hardware to work in *three modes of operation*: real-time measurement mode, sweeping measurement mode and hybrid measurement mode.

The *real-time measurement mode* supports real-time measurements on 25 MHz of bandwidth at most. The samples gathered from the USRP2 hardware are post-processed in the GNU radio host computer environment, employing FFT (Fast Fourier Transform) analysis of the received data. This allows various frequency resolutions, but high FFT size values (higher than 1024) are not recommended since the number of operations to calculate FFT transform increases exponentially as the FFT size rises. This can cause disturbance of the operation of the USRP2 due to the higher processing requirements and the inability of the host computer to process in-time all data coming from the USRP2. Before the actual FFT analysis of the samples, time windowing is performed in order to reduce the spectral leakage (a side-effect coming from the time restrictions). The current implementation uses the Hamming Window. In addition, FFT overlapping is used and the overlapping frequency points are dropped to reduce the non linear response of the DDC at the edges of the FFT analysis. The real-time measurement mode utilizing FFT is consisted of five processing blocks (Figure 1) This mode of operation can be also performed without FFT analysis of the data, excluding the use of *gr_fft_vcc* block.

The *sweeping mode of operation* of the sniffer enables particular values for the resolution bandwidth in the range 195.3125 KHz – 25 MHz, corresponding to the respective decimation factors of the digital-down-conversion process. Here, the USRP2 periodically switches the center frequency between sequential frequency bands, with the chosen resolution bandwidth size, to cover the full requested frequency span. This mode of operation is employed when the required frequency span is higher than the largest possible receiving bandwidth size of 25 MHz. When using this mode of operation, only four blocks are connected in the flow graph: *usrp2_source_32fc*, *gr_stream_to_vector*, *gr_complex_to_mag_squared* and *gr_energy_detector_f*, where the *gr_energy_detector_f* initiates the switching between the sequential frequency bands, when the dwell time at a frequency band is expired.

The sweeping mode of operation can be jointly combined with FFT analysis to multiplicatively increase the resolution bandwidth of the USRP2 hardware, resulting in *hybrid mode of operation* of the sniffer. This mode of operation can offer higher sensing performances due to the increased frequency resolution as well as reduced sweep time requirements. When targeting a particular resolution bandwidth size, the number of sweeps (consequently the sweep time) to cover the full span of interest can be reduced by the usage of the highest receiving bandwidth (lowest possible decimation factor) and specific FFT size to support the resolution bandwidth requirement. This mode of operation utilizes five processing blocks (Figure 1): *usrp2_source_32fc*, *gr_stream_to_vector*, *gr_fft_vcc*, *gr_complex_to_mag_squared* and *gr_energy_detector_f*, where the last is in charge of initiating the switching between frequency bands.

The following section presents several sensing related applications of the previously elaborated USRP2 sniffer implementation. It shows its versatility and potentials for practical usage.

III. USRP2 SNIFFER APPLICATIONS

The basic application of the developed USRP2 based sniffer implementation is in the area of assisting the dynamic spectrum access process, i.e., to serve for detection and localization of secondary spectrum access opportunities. Several applications engaging the USRP2 based sniffing have been included in part of the authors' previous work. The applications are classified in two types: *energy detector based applications* and *HOS detector based applications*, each elaborated in more details in the subsequent subsections.

A. *Applications based on energy detector*

The classical energy detection technique employed by the USRP2 based sniffer implementation is consisted of the minimum hold, maximum hold and the mean hold detector types. This subsection presents two applications of the USRP2 based energy detector, the first focusing on frequency spectrum measurements, the second on data fusion and dynamic radio environmental maps (REMs) derivation.

1) Medium and long term spectrum measurements

The classical energy detector in the USRP2 sniffer implementation can be included in medium and long term spectrum measurement campaigns. Wideband measurements can mainly employ the sweeping and hybrid mode of operation of the sniffer, while band–specific measurements can be performed with the real-time measurement mode of the sniffer (if the bandwidth of interest does not surpass the USRP2 hardware limitations, i.e., the maximal bandwidth of 25 MHz).

Ref. [11] shows the usage of the USRP2 sniffer implementation for the 2.4 GHz ISM band inspection. The measurement setup includes the USRP2 hardware comprising the RFX2400 daughterboards. The focus on the campaign was on the definition of the measurement methodology for the referred frequency band. The USRP2 based sniffer implementation is used mainly in the sweeping and hybrid operational mode with maximum hold energy detection type. The results prove that this USRP2 sniffer implementation offers sensing and detection performances comparable to high end-devices performances. Moreover, it has been verified that the hybrid mode of operation of the sniffer offers significantly better performances than the classical sweeping operational mode (Figure 2 [11]).

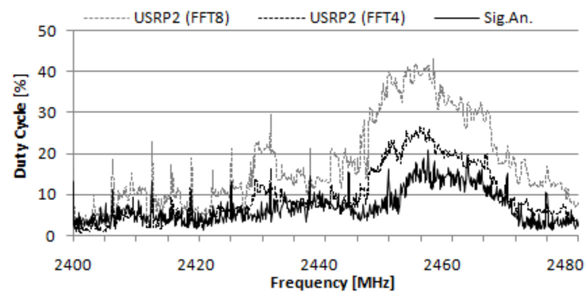

Figure 2. 2.4 GHz ISM duty cycle results of the USPR2 sniffer in hybrid mode and spectrum analyzer, RBW =100KHz and sweep=1s

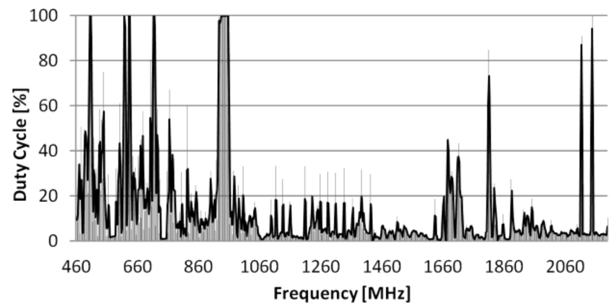

Figure 3. Wideband (460–2200 MHz) duty cycle results in working day busy hour (12:00–18:00 hours) gathered using the USPR2 sniffer in hybrid mode

Wideband measurements have been also performed with the energy detector of the USRP2 based spectrum sniffer implementation using WBX daughterboards [7]. The USRP2 sniffer implementation is used in hybrid operational mode with mean hold detection for these measurements employing FFT with size 16 and receiving bandwidth of 1 MHz, resulting in resolution bandwidth of 62.5 KHz. Figure 3 plots the duty cycle results for the spectrum usage gathered in this campaign.

B. *Dynamic Radio Environmental Maps Creation*

Additional application of classical energy detection option of the USRP2 spectrum sniffer implementation can be to provide input to the process of dynamic derivation of radio environmental maps (REMs). The USRP2 sniffers can be distributed on various locations in the observed environment and can report received power levels at specific frequency bands to a centralized fusion center. The fusion center gathers the spectrum data from the scattered USRP2 based sniffers and combines the information into REMs. This process can be performed in a dynamic fashion and can impact the discovery of the spatial distribution of the primary users in cognitive network. Therefore, it can facilitate the secondary system spectrum opportunities detection.

Figure 4 depicts the results of a real-time REM calculation demo. The demo scenario consists of two WLAN signal sources, one access point and one laptop computer as a source. Four USRP2 sniffers are placed in an indoor environment and a fusion center dynamically interpolates (with modified IDW interpolation [12]) the data

 ISBN: 978-1-61208-131-1

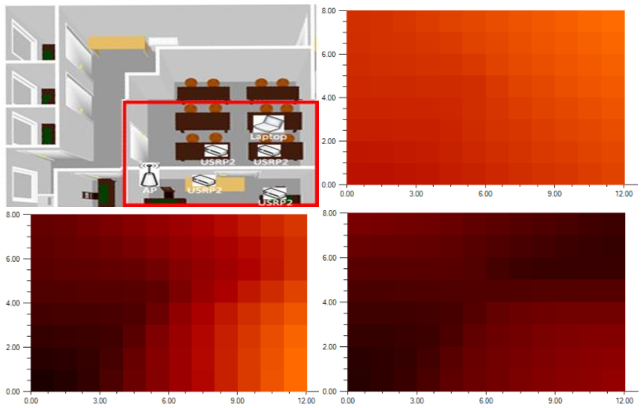

Figure 4. Real-time REM calculation demo: no source (upper right plot), one source (lower left plot), two sources (lower right plot)

from the scattered USRP2 sniffers. As seen on Figure 4, the interpolation data is able to roughly localize both signal sources.

C. Applications based on HOS detector

This part presents two applications of the HOS detector option of the USRP2 sniffer implementation in cognitive environments, i.e., a *channel selection application* and *cooperative channel selection application*. Both applications utilize the USRP2 sniffer implementation in sweeping measurement mode with HOS detection type.

1) Channel selection based on HOS detection

The HOS detector of the USRP2 sniffer implementation can be useful in cognitive radios in order to serve the channel selection process. Namely, the consideration of the higher order statistics values in the channel selection improves the decisions for the best channel. The impact of the HOS detection has been tested on USRP2 devices in a demo based on real environment conditions. The channels width in the demo is chosen to be 2.5 MHz and the center frequencies of the channels are chosen in the manner to include the WiFi channels center frequencies. The demo is focused on the full 2.4GHz ISM band (due to hardware limitations, RFX2400 daughterboards) and Figure 5 plots the environment conditions during the tests. The channel classification criteria is based on average, skewness and kurtosis values of the received power, each of the referred considered with utility factors *a*, *b* and *c*, respectively. The target is to select a channel with statistics closest to the system noise. Figure 6 plots the dependence of the channel selection probability and the utility factors settings. As can be noticed (considering the environment conditions), the worse results are gained when the channel selection depends mostly on the average received power. Predefined frequencies are chosen due to the USRP2 hardware non-idealities – it has different noise power levels and different variations at different frequencies. The decisions on the best channel are more confident when the channel decision depends more on the skewness and kurtosis of the received power. The best performances in terms of channel selection are gained when the decision depends equally on the skewness and kurtosis of the received power, i.e., *b*=c=0.5. This result reflects the actual environment conditions (illustrated on Figure 5).

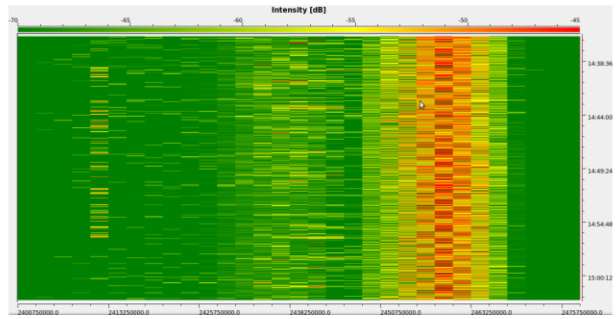

Figure 5. 2.4 GHz ISM spectrum conditions during USRP2 HOS detector demo

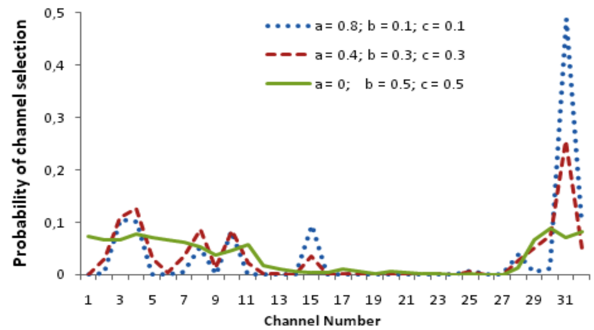

Figure 6. HOS channel selection based on different utility factors settings

The dependence of the channel selection with the sensing time duration is presented in Table I. Here, a channel is considered as free if the duty cycle of the channel occupancy is lower than 5% (5 dB above noise level criteria, inspection time 30 min), and the maximum received energy does not exceed 10 dB above average noise (the most green channels in Figure 5). The channel selection decision improves as the sensing period is higher and sensing period of 3.2 s yields a free channel selection probability of 1 (thus resulting in successful avoidance of busy channels, Table I).

TABLE I. HOS DETECTION: CHANNEL SELECTION PROBABILITY

Sensing period	Busy channel selection probability	Free channel selection probability
800 ms	0.226037	0.773963
1600 ms	0.118825	0.881175
3200 ms	0	1

2) Cooperative channel selection based on HOS data

The HOS detection option of the USRP2 based sniffer implementation has also been included in a cooperative demo comprising USRP2 based cognitive nodes. The cooperative nodes exchange their mean power, skewness and

 ISBN: 978-1-61208-131-1

kurtosis values in distributed fashion according to the RAC^{2}E [13] rendezvous protocol for cognitive ad-hoc networks. After a simple fusion (with averaging) of the gathered data, a source USRP2 based cognitive radio finds the best channel available and starts the communication with a destination USRP2 node. The targeted band in the cooperative demo is the 2.4 GHz ISM band and the environment conditions are the same as in the demo presented in the previous part (the tests were run simultaneously). The demo aims to prove how the cooperative exchange and fusion of the HOS data among the cooperative nodes improves the best channel decision.

TABLE II. COOPERATIVE HOS DETECTION: CHANNEL SELECTION PROBABILITY

Sensing period	Cooperative nodes	Busy channel selection probability	Free channel selection probability
800 ms	1	0.226037	0.773963
	2	0.138968	0.861032
	3	0.071736	0.928264
1600 ms	1	0.118825	0.881175
	2	0.029412	0.970588
	3	0.014706	0.985294

The results (Table II) prove that, as the number of cooperative nodes increases, the channel selection process is more reliable (free channels are most probably to be chosen, whereas busy channels are effectively avoided). As can be noticed, with sensing period of 1600 ms and three cooperative nodes, the probability of selection of a busy channel is 1.5%, and the probability of selection of a free channel is 98.5%.

IV. CONCLUSION AND FUTURE WORKS

Future wireless networks will focus on dynamic and flexible usage of spectrum resources. The importance of spectrum sensing is crucial in such cognitive environments. It provides support for spectrum awareness and dynamic spectrum opportunities detection. Therefore, providing simple and cheap spectrum sensing solutions with good accuracy is an important research challenge. The mid-end devices can be an efficient tool for spectrum sensing, but must be carefully configured and used depending on the actual application.

This paper presents a flexible and efficient USRP2 based spectrum sniffer implementation. The main advantages of the developed C++ sniffer implementation are the support of different modes of operation and detection types, as well as better timing performances than the original python based GNU radio implementation. This enables the developed sniffer implementation to be implemented in various cognitive radio applications. Several of them were presented in the paper and they demonstrate the applicability potential of the USRP2 sniffer implementation in cognitive environments.

Future work will focus on additional detector types implementation, as well as comparative analysis of the different detectors' performances. The spectrum sniffer will be implemented in additional cognitive radio applications, i.e. different cooperative sensing and fusion methods can be tested in applications comprising the referred sniffer implementation. Furthermore, the USRP2 sniffer implementation can be included in derivation of secondary spectrum usage strategies, such as spectrum driven policy derivation and integration into a policy based system [14].

ACKNOWLEDGMENT

Parts of this work were funded by the EC through the FP7 project FARAMIR (248351) [15]. The authors would like to thank everyone involved.

REFERENCES

[1] T. Ycek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 1, pp. 116–160, 2009.

[2] M. Lopez-Benítez et al, "Performance of spectrum sensing for cognitive radio based on field measurements of various radio technologies," in Proceedings of the *16th European Wireless Conference (EW) 2010, Special session on Cognitive Radio*, Lucca, Italy, April 12-15, 2010.

[3] M. Wellens and P. Mähönen, "Lessons Learned from an Extensive Spectrum Occupancy Measurement Campaign and a Stochastic Duty Cycle Model", *Springer Mobile Networks and Applications*, 2009.

[4] Z.Quan et al, "An Optimal Strategy for Cooperative Spectrum Sensing in Cognitive Radio Networks", *IEEE Global Communications Conference (GLOBECOM) 2007*, Washington, DC, USA, November, 2007.

[5] J. Unnikrishnan et al, "Cooperative Spectrum Sensing and Detection for Cognitive Radio", *IEEE Global Communications Conference (GLOBECOM) 2007*, Washington, DC, USA, Nov. 2007.

[6] R. Vuyyuru et al, "Evaluation of Energy Based Spectrum Sensing Algorithm for Vehicular Networks, SDR Forum Technical Conference 2009, December 2009.

[7] Universal Software Radio Peripheral 2 (USRP2). Information available at: www.ettus.com.

[8] GNU Radio software development toolkit. Information available at: http://gnuradio.org/redmine/wiki/gnuradio.

[9] D. N. Joanes, C. A. Gill, " Comparing measures of sample skewness and kurtosis", *Journal of the Royal Statistical Society: Series D (The Statistician)*, Volume 47, Issue 1, pages 183–189, April 1998.

[10] Z. Chen et al, "Demonstration on Real time Spectrum Sensing for Cognitive Radio", *IEEE Communications Letters*, vol.14, no.10,2010.

[11] D. Denkovski, M. Pavloski, V. Atanasovski and L. Gavrilovska, "Parameter settings for 2.4GHz ISM spectrum measurements," *3rd International Symposium on Applied Sciences in Biomedical and Communication Technologies (ISABEL) 2010*, Rome, Italy, 7-10 November, 2010.

[12] R. J. Renka , "Multivariate interpolation of large sets of scattered data," ACM Transactions on Mathematical Software 14, 2, 139–148, 1988.

[13] V. Pavlovska, D. Denkovski, V. Atanasovski and L. Gavrilovska, "Novel Rendezvous Protocol for Asynchronous Cognitive Radios in Cooperative Environments*", the 21st Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC) 2010*, Istanbul, Turkey, 26-30 September, 2010.

[14] D. Denkovski, V. Pavlovska, V. Atanasovski and L. Gavrilovska, "Novel Policy Reasoning Architecture for Cognitive Radio Environments", *IEEE Global Communications Conference (GLOBECOM) 2010*, Miami, USA, 6-10 December, 2010.

[15] EC FP7 project FARAMIR, Information available at: http://www.ict-faramir.eu.