Implementation of an adaptive OFDMA PHY/MAC on USRP platforms for a cognitive tactical radio network

Vincent Le Nir and Bart Scheers
Department Communication, Information Systems and Sensors (CISS)
Royal Military Academy (RMA)
30, Avenue de la Renaissance
B-1000 Brussels, Belgium
E-mail: {vincent.lenir}{bart.scheers}@rma.ac.be

***Abstract*—Cognitive radio is envisioned to solve the problem of spectrum scarcity in military networks and to autonomously adapt to rapidly changing radio environment conditions and user needs. Dynamic spectrum management techniques are needed for the coexistence of multiple cognitive tactical radio networks. Previous work has investigated the iterative waterfilling algorithm (IWFA) as a possible candidate to improve the coexistence of such networks. It has been shown that adding a constraint on the number of transmitter's sub-channels improves the convergence of IWFA. In this paper, we propose an adaptive orthogonal frequency division multiple access (OFDMA) physical (PHY) and medium access control (MAC) for the coexistence of multiple cognitive tactical radio networks. The proposed scheme is implemented on universal software radio peripheral (USRP) platforms using Qt4/IT++ and the USRP hardware driver (UHD) application programming interface (API).**

I. INTRODUCTION

Cognitive radio (CR) has been introduced by Mitola as an extension of software radio. In this technology, radio nodes are intelligent agents that search out ways to deliver services according to the user needs and the radio environment [1]. CR has been an active topic of research since most regulatory bodies found that the spectrum is underutilized although most available spectrum is licensed, leaving small room for future wireless applications [2].

CR can also solve the problem of spectrum scarcity for military networks. As the electro-magnetic environment in an operational theater can be very hostile, cognitive tactical radio networks can autonomously adapt to rapidly changing conditions and user needs [3].

The coexistence of cognitive tactical radio networks requires dynamic spectrum management techniques in order to reduce the interference and to improve the performance in terms of throughput, power (longer battery life), and delay. Dynamic spectrum management techniques can be centralized/distributed (decisions are made centrally/locally), cooperative/non-cooperative (some information is shared/not shared between networks), and can use an horizontal/vertical sharing model (all networks have equal/limited rights to access the spectrum) [4]. Our target application for the coexistence of cognitive tactical radio networks corresponds to a mission involving multiple coalition nations in a foreign country with no a priori frequency planning, meaning that there is no frequency management cell (FMC) to coordinate the different networks. Moreover, the networks can't exchange information between each others and they have equal priority rights. Therefore, we are interested in a distributed non-cooperative technique in an horizontal spectrum sharing model.

The iterative waterfilling algorithm (IWFA) is an adequate dynamic spectrum management technique to meet these requirements [5]. Indeed, IWFA is an autonomous algorithm solving the distributed power control problem in a frequency selective interference channel. Robust versions of the IWFA have been designed to cope with dynamic wireless channels [6], [7], [8], [9], [10]. However, IWFA does not converge to a unique solution (multiple Nash equilibria). This aspect is inherent to IWFA because at each iteration some power is poured in the best sub-channel regardless the interference caused to the other networks, while they have a better benefit avoiding each other by taking different sub-channels. The convergence of the IWFA can be improved by adding a constraint on the number of transmitter's sub-channels [11], [12].

Random access multi-channel medium access control (MAC) protocols have been proposed for CR networks [13]. Multi-channel MAC protocols have also been designed for IWFA [6], [14]. These protocols use a dedicated control channel to coordinate the radios. It is unlikely that a dedicated control channel can be used in a military context since it is a single point of failure. A possible workaround is to use a rendezvous multi-channel MAC protocol, however the radios may beacon for a long time before establishing a rendezvous. Another alternative is to employ time division multiplexing access (TDMA) to obtain a collision-free transmission schedule or frequency division multiplexing access (FDMA) as described in [15]. In this paper, we propose an adaptive orthogonal frequency division multiple access (OFDMA) PHYsical/MAC to allow simultaneous collision-free transmissions in a cognitive tactical radio network. The adaptive OFDMA PHY/MAC has the following characteristics:

- It is based on the IWFA with selection of a single sub-channel [11], [12]. It consists of grouping several OFDM sub-carriers to form a sub-channel. In a first mode, it uses a fixed bit-loading per sub-carrier. In a second mode, it uses an adaptive bit-loading related to the spectrum sensing and the channel estimation on each OFDM sub-carrier.
- It is robust against multipath due to the insertion of a cyclic prefix and allows a single-tap equalizer due to the orthogonality of the sub-carriers.
- It uses a blind demodulation chain, meaning that it uses the cyclic prefix to estimate blindly the timing offset and to detect the presence of an OFDM signal. The timing offset estimate is used to synchronize control packets and data packets, and to estimate blindly the frequency offset, transmission channel, phase offset and transmitted bits. The residual ambiguity given by the BPSK or the adaptive QAM modulation schemes is solved by transmitting a single sub-carrier pilot or by using differential encoding.

The adaptive OFDMA PHY/MAC has been implemented on universal software radio peripheral (USRP) platforms using Qt4/IT++ and the USRP hardware driver (UHD) application programming interface (API). Qt is a cross-platform application framework that is widely used for developing application software with a graphical user interface (GUI) [16]. IT++ is a C++ library of mathematical, signal processing and communication classes and functions. Its main use is in simulation of communication systems and for performing research in the area of communications [17]. The goal of the UHD is to provide a host driver and API for current and future USRP products. The UHD driver can be used standalone or with 3rd party applications such as Gnuradio, Labview, or Simulink [18].

This paper is organized as follows. First, the adaptive OFDMA MAC protocol is described in Section II. Second, the adaptive OFDMA PHysical layer functions are described in Section III. The implementation of the adaptive OFDMA PHY/MAC on USRP platforms using Qt4/IT++ and the UHD API is described in Section IV. Finally, Section V concludes the paper.

II. Adaptive OFDMA MAC protocol

In this Section, the adaptive OFDMA MAC protocol is described. It is assumed that the CRs have agreed on the same front-end parameters to transmit and receive, i.e. carrier frequencies, sampling rates and bandwidths. It is also assumed that the CRs have the same OFDM parameters, i.e. number of subcarriers, cyclic prefix (CP) size, and number of sub-channels. The adaptive OFDMA MAC protocol uses control packets for the handshaking between two CRs. These control packets are request-to-send (RTS) and clear-to-send (CTS) packets. In a first mode, the two control packets include source address, destination address and best sub-channel sensed by the CR. In a second mode, control packets also include target rate constraint, bit and power allocation. Therefore, unlike carrier sense multiple access (CSMA) scheme and other random access multi-channel MAC protocols, the two control packets convey some information and need to be aligned with the timing of OFDM symbols transmitted in the same bandwidth of interest to keep the orthogonality between sub-carriers.

A. *First Mode*

The first mode uses a fixed bit-loading, e.g. a fixed BPSK or QAM modulation over the different sub-carriers. In the following, we suppose a transmission from CR 1 to CR 2. CR 1 and CR 2 perform spectrum sensing in the bandwidth of interest and determine their best sub-channel based on an energy estimate. The RTS includes the source address CR 1, the destination address CR 2, and the best sub-channel of CR 1 given by spectrum sensing. CR 1 transmits the RTS on its best sub-channel. CR 2 demodulates all the sub-channels in parallel (see Section 3.3) and discovers that a RTS is sent based on the destination address of CR 1. CR 2 gets the RTS source address and the best sub-channel of CR 1. The CTS includes the CR 2 source address, the CR 1 destination address, and the best sub-channel of CR 2 by spectrum sensing. CR 2 transmits the CTS on CR 1 best sub-channel. CR 1 knows to receive on its best sub-channel. CR 1 demodulates the CTS and gets CR 2 best sub-channel. Finally, CR 1 transmits the data on CR 2 best sub-channel.

B. *Second Mode*

The second mode uses an adaptive QAM over the different sub-carriers. CR 1 and CR 2 determine their best sub-channel based on an energy estimate. The RTS includes the CR 1 source address, the CR 2 destination address, the CR 1 best sub-channel, and the target rate constraint. CR 1 transmits the RTS on its best sub-channel. CR 2 demodulates all the sub-channels in parallel and discovers that a RTS is sent based on the destination address of CR 1. CR 2 gets the RTS source address, the target rate constraint, and the best sub-channel of CR 1. If the CR 1 best sub-channel is the same as the CR 2 best sub-channel, the CTS includes the CR 2 source address, the CR 1 destination address, the CR 2 best sub-channel, and the bit and power allocation for CR 1 target rate constraint. CR 2 transmits the CTS on CR 1 best sub-channels. CR 1 knows to receive on its best sub-channel. CR 1 demodulates the CTS and gets CR 2 best sub-channel, as well as the bit and power allocation for its target rate constraint. Finally, CR 1 transmits the data on their common best sub-channel with adaptive QAM. If the CR 1 best sub-channel is different from the CR 2 best sub-channels, CR 1 gets the best sub-channel of CR 2 by the CTS, and send a second RTS on CR 2 best sub-channel to allow CR 2 to compute the bit and power allocation. CR 2 sends a second CTS with this information on CR 1 best sub-channel. Finally, CR 1 transmits the data on CR 2 best sub-channel with adaptive QAM.

III. Adaptive OFDMA PHYsical layer

In this Section, the key functions of the adaptive OFDMA PHY are described, i.e. the spectrum sensing, the OFDM signal detection, the blind OFDM demodulation and the distributed bit and power allocation.

A. Spectrum sensing

A CR needs to determine its best sub-channel and communicate this information to the another CR. The bandwidth of the signal of interest B is divided into a number of sub-channels S of width B/S. As there is no assumption about the type of the observed signals, a nonparametric method should be used. We propose to use the classical Barlett's method for power spectrum estimation known also as the method of averaged periodograms [19]. We assume that the complex sampling rate equals the bandwidth of the signal of interest. The averaged periodograms for N sub-carriers of a complex baseband signal with K blocks of N samples $\mathbf{y} = [y(kN), \ldots, y((k+1)N-1)]$ with $k = [0, \ldots, K-1]$ is given by

$$E(i) = \frac{1}{KN} \sum_{k=0}^{K-1} | \sum_{n=0}^{N-1} y(kN+n) e^{-\frac{j2\pi in}{N}} |^2 \quad (1)$$

with $i = [0, \ldots, N-1]$ bins. The best sub-channel selection consists of integrating the estimated power spectrum of the frequency bins corresponding to the width B/S for all sub-channels and determine the sub-channel which have the lowest energy

$$S_{opt} = \underset{m}{min} \sum_{i=\frac{mN}{S}}^{\frac{(m+1)N}{S}-1} E(i) \quad (2)$$

with $m = [0, \ldots, S-1]$. This sensing procedure is used first by CR 1 for the selection of its best sub-channel for RTS transmission, and is further embedded in the RTS control packet. Secondly, this sensing procedure is used by CR 2 to communicate its best sub-channel to CR 1 in the CTS control packet via CR 1 best sub-channel.

B. OFDM Signal Detection

The CR also needs to detect within the bandwidth of interest B the presence of an OFDM signal even if the signal is sent over a single sub-channel. A survey of OFDM signal detection techniques can be found in [20], [21], [22], [23]. Some techniques require the knowledge of a pilot sequence, OFDM parameters, and noise variance to determine the detection threshold. Detection techniques requiring the knowledge of a pilot sequence have better detection performance, but worse bandwidth/power/complexity efficiency. Detection techniques requiring the knowledge of the noise variance assumes a white noise assumption and degrade severely in the presence of colored noise. Moreover, when the theoretical values of the threshold are not known, they have to be empirically computed by feeding the detector with pure noise signals and calculating the test statistic.

In the proposed adaptive OFDMA MAC protocol, control and data packets are sent only over a subset of the available sub-carriers. Moreover, other unknown signals or colored noise can be present in the bandwidth of interest. We propose to use a modified version of the cyclic prefix based sliding window correlation detector to determine the presence of an OFDM signal in the bandwidth of interest. This detector does not require the knowledge of a pilot sequence but only requires the knowledge of the OFDM parameters, i.e. number of subcarriers N, cyclic prefix size P. The detection threshold is based on the non-correlated part of the cyclic prefix sliding window correlation estimate instead of the noise variance. This allows to detect the presence of an OFDM signal even in the presence of an unknown signal in the bandwidth of interest. Moreover, this detector also gives an estimate of the OFDM timing offset when it is detected. Assuming a complex baseband signal with K blocks of $N+P$ samples $\mathbf{y} = [y(k(N+P)), \ldots, y((k+1)(N+P)-1)]$ with $k = [0, \ldots, K-1]$, the cyclic prefix sliding window correlation estimate is given by

$$\rho(\theta) = \frac{| \sum_{k=0}^{K-2} \sum_{j=\theta}^{\theta+P-1} y(k(N+P)+j) y^*(k(N+P)+j+N)|}{(K-1)P\sigma_y^2} \quad (3)$$

with σ_y^2 the variance of the received complex baseband signal. The absolute value of the correlation estimate is able to cope with frequency and phase offsets introduced by Doppler shifts and clock mismatches. The estimate of the timing offset is given by

$$\theta^{opt} = \underset{\theta \in \{0, \ldots, N+P-1\}}{max} \rho(\theta) \quad (4)$$

In order to determine the presence of an OFDM signal, the estimate of the timing offset is compared with the non-correlated part of the cyclic prefix sliding window correlation estimate. Assuming a channel delay spread spanning the entire cyclic prefix, the non-correlated part of the cyclic prefix sliding window correlation estimate is given by $\rho(mod(\theta^{opt}+2P, N+P))$. Assuming that the non-correlated part is a Gaussian distribution with mean m_{nc} and variance σ_{nc}^2, the detection threshold η is given by

$$\eta = m_{nc} + \alpha\sigma_{nc} \quad (5)$$

with α an integer corresponding to the number of standard deviations necessary to discriminate between an OFDM signal and a non-OFDM signal using only the cyclic prefix sliding window correlation estimate $\rho(\theta)$. The detector scheme is

$$\begin{array}{ll} \rho(\theta^{opt}) > \eta & \text{Presence of an OFDM signal} \\ \rho(\theta^{opt}) < \eta & \text{Absence of an OFDM signal} \end{array} \quad (6)$$

Using $K = 50$ blocks in the application, α has been set to 40, allowing very few false alarms and misdetections.

C. Blind OFDM demodulation

The cyclic prefix sliding window correlation estimate can be used to estimate the frequency offset [24]

$$\epsilon^{opt} = -\frac{1}{2\pi N} \angle \rho(\theta^{opt}) \quad (7)$$

After time and frequency offset corrections, the OFDM symbols are transformed to the frequency domain by the discrete Fourier transform (DFT) operation. As there is no interference between two consecutive OFDM symbols, we obtain independent subcarriers with the following channel model

$$Y(kN+i) = H(kN+i)X(kN+i) + N(kN+i) \quad (8)$$

with $k = [0, \ldots, K-1]$ and $i = [0, \ldots, N-1]$, in which $Y(kN+i)$, $H(kN+i)$, $X(kN+i)$, and $N(kN+i)$ are respectively the demodulated data, the channel frequency response, the transmitted symbol and the noise for the block k and sub-carrier i. Assuming the channel invariant over the K blocks, a blind estimate of the channel amplitude is given by

$$|H(i)^{opt}|^2 = \frac{1}{K} \sum_{k=0}^{K-1} |Y(kN+i)|^2 \quad (9)$$

Assuming the channel invariant over the K blocks, a blind estimate of the phase offset can be obtained for M-PSK signals [25] by the following expression

$$\phi(i)^{est} = \angle \frac{1}{K} \sum_{k=0}^{K-1} Y(kN+i)^M \quad (10)$$

Modulation stripping has also been investigated for M-QAM signals in [26]. Knowing the phase offset estimate for each sub-carrier, it is possible to correct linear shift of the phase in the frequency domain due to an incorrect timing offset estimate belonging to the ISI free region, as well as abrupt changes of the phase in the frequency domain due to the phase ambiguity introduced by the blind phase offset algorithm (phase unwrapping). The phase correction for M-PSK signals is performed using the Algorithm 1.

Algorithm 1 Phase correction for M-PSK signals

$\phi(0)^{opt} = \phi(0)^{est}$
for j=1 to N-1
  if $|\phi(i)^{est} - \phi(i-1)^{est}| > \frac{\pi}{M}$
  then $\phi(i)^{opt} = |\phi(i)^{est} - \phi(i-1)^{est}| - \frac{2\pi}{M}$
  if $|\phi(i)^{est} - \phi(i-1)^{est}| < \frac{\pi}{M}$
  then $\phi(i)^{opt} = \phi(i)^{est} - \phi(i-1)^{est}$
  $\phi(i)^{opt} = \phi(i)^{opt} + \phi(i-1)^{opt}$
end for

A differential encoding of the data or a single pilot can be used to remove the M-fold phase ambiguity and retrieve the transmitted bitstream.

Algorithm 2 Distributed adaptive bit and power allocation

Initalisation $\lambda = 10^{-9}$, $\Gamma = 9.8$ dB, $R^{target} = 64$ kbps
repeat
  repeat
    for $i = \frac{S^{opt}N}{S}$ to $\frac{(S^{opt}+1)N}{S} - 1$
      $p(i) = \left[\frac{1}{\lambda \log_2(2)} - \frac{\Gamma E(i)}{|H(i)^{opt}|^2}\right]^+$
      $b(i)^{opt} = \lfloor \log_2(1 + \frac{|H(i)^{opt}|^2 p(i)}{\Gamma E(i)}) \rfloor$
      $p(i)^{opt} = 2^{b(i)^{opt}-1} \frac{\Gamma E(i)}{|H(i)^{opt}|^2}$
    end for
    if $\sum_{i=\frac{S^{opt}N}{S}}^{\frac{(S^{opt}+1)N}{S}-1} p(i)^{opt} < P^{tot}$ decrease λ
    if $\sum_{i=\frac{S^{opt}N}{S}}^{\frac{(S^{opt}+1)N}{S}-1} p(i)^{opt} > P^{tot}$ increase λ
  until the desired accuracy is reached
  Calculate $R = \Delta f \sum_{i=\frac{S^{opt}N}{S}}^{\frac{(S^{opt}+1)N}{S}-1} b(i)^{opt}$
  if $R < R^{target}$ increase P^{tot}
  if $R > R^{target}$ decrease P^{tot}
until the desired accuracy is reached

D. Distributed adaptive bit and power allocation

The distributed bit and power allocation (adaptive QAM) can only be determined when a CR sends its control packet on the other CR best sub-channel. If CR 1 best sub-channel is the same as CR 2 best sub-channel, then only one handshake is necessary (RTS-CTS) to inform CR 1 about CR 2 best sub-channel, bit and power allocation. However, if CR 1 best sub-channel is different from CR 2 best sub-channel, two handshakes are necessary (RTS-CTS-RTS-CTS) because CR 2 informs CR 1 about its best sub-channel in the first CTS and the bit and power allocation in the second CTS. The distributed bit and power allocation is based on the waterfilling algorithm, in which an inner loop maximizes the bit allocation for a transmit power constraint, and an outer loop minimizes the transmit power for a target rate constraint. The algorithm for distributed bit and power allocation is described in Algorithm 2, in which λ is the Lagrangian parameter [27], Γ is the SNR gap which measures the loss with respect to theoretically optimum performance [28], $\mathbf{p} = [p(0), \ldots, p(N-1)]$ and $\mathbf{p}^{opt} = [p(0)^{opt}, \ldots, p(N-1)^{opt}]$ are the power allocation vectors, $\mathbf{b}^{opt} = [b(0)^{opt}, \ldots, b(N-1)^{opt}]$ is the bit allocation vector, P^{tot} is the total power constraint, Δf is the sub-carrier bandwidth, R is the data rate and R^{target} is the target rate constraint. As shown in [11], [12], when the number of sub-channels is lower than the number of concurrent links, CR users have to share the same sub-channels and iterative updates of bit and power allocation can lead to convergence problems due to multiple Nash equilibria. To avoid this problem, the number of sub-channels is larger than the number of concurrent links to ensure the convergence to a single Nash equilibrium. This leads to an distributed adaptive OFDMA solution.

IV. Implementation using Qt4/IT++ and the UHD API

In this Section, the implementation of the adaptive OFDMA PHY/MAC on USRP platforms using Qt4/IT++ and the UHD API is described. Several classes and procedures have been implemented for this application using QThread to enable multi-threading, Qwt to enable plotting, IT++ to enable mathematical operations, and Gstreamer to enable video/audio transmission and reception.

A. Names and description of the implemented classes

- Class BitWaterfilling. This class does the bit and power allocation according to spectrum sensed and estimated channel for a target rate constraint or a total power constraint.
- Class BlindOFDM. This class gets a vector of bits from a named pipe (FIFO) coming from text, video, or audio in the application. It modulates this vector of bits into a fixed BPSK, QAM or an adaptive QAM OFDM vector according to its best group of subcarriers. It also performs a blind detection of a received OFDM vector based on the cyclic prefix, and blindly demodulates a received fixed BPSK, QAM or an adaptive QAM OFDM vector into a vector of bits according to its best group of subcarriers (blind time offset correction, blind frequency offset correction, blind phase offset estimation). Finally, it puts a vector of bits into a named pipe (FIFO) to be read by a text reader, video reader, audio reader in the application.
- Class File. This class converts from/to characters to/from bits. It also reads/writes a vector of bits from/to a file.
- Class MainWindow. This class is dedicated to the GUI thread. This class has a first push Button to Start/Stop Tx, a second push Button to Start/Stop Rx, and a third push Button to Start/Stop Video. It uses a lineEdit to take command or to input text and textEdit to display some text. It also uses multiple lineEdits to control Tx rate, Tx frequency, Tx gain, Tx amplitude, Rx rate, Rx frequency, Rx gain, FFT size, CP size, and number of sub-channels.
- Class Packets. This class does a conversion between vector of a double, float, integer and a vector of bits to include in the RTS/CTS packets. It encodes/decodes the RTS/CTS packets with necessary information (source address, destination address, best sub-channel, bit and power allocation).
- Class Plot. This class plots some information (spectrum sensing, best group of sub-channels, target rate constraint, bit and power allocation).
- Class Protocols. This class is dedicated to the worker thread (QThread). It can reinitialize the parameters as requested by the GUI (FFT size, CP size, number of sub-channels). This class implements a state machine for a Tx/Rx handshaking adaptive OFDMA MAC protocol based on RTS sending/listening, CTS listening/sending and data sending/listening. It also performs a time offset estimation before RTS sending, CTS sending to avoid inter-carrier interference.
- Class Sensing. This class estimates the spectrum based on averaged FFT. It takes a decision based on the mean spectrum estimated shifted by a fixed amount of dB, and selects the best group of sub-carriers according to the number of sub-channels.
- Class Text. This class is implemented as a separate thread (QThread). It reads some input text in the GUI and put it in a named pipe (FIFO). It also displays some text in the GUI from a named pipe (FIFO).
- Class UHDDevice. This class reinitializes USRP parameters as requested in the GUI (Tx rate, Tx frequency, Tx gain, amplitude, Rx rate, Rx frequency, Rx gain). It reads a certain amount of samples at a particular time (timestamp). It writes a certain amount of samples at a particular time (timestamp). It also checks for errors after sending/receiving data. The procedures are implemented such that the received and transmitted samples are in a steady state for the USRP (first samples should be discarded because of the time to power up).
- Class Video. This class is implemented as a separate thread (QThread). It uses a Gstreamer transmit pipeline showing the video transmitted on screen and writing the same data in a named pipe (FIFO) which will be read by the OFDM modulator for transmission. It also uses a Gstreamer receive pipeline showing the video received on screen if a video container is detected.

B. Description of the application

Figure 1 shows the application window for spectrum sensing. The GUI parameters can be chosen during run-time. The transmission parameters are the Tx Rate in Msps, the Tx Frequency in MHz, the Tx Gain in dB, the Tx Amplitude corresponding to a linear multiplication factor. The reception parameters are the Rx rate in Msps, the Rx frequency in MHz, the Rx Gain in dB. The OFDM parameters are the FFT size, the CP size, and the number of sub-channels. There are two buttons to Start Tx and to Start Rx, which correspond to different state machines. The Tx state machine does a spectrum sensing, send RTS, receive CTS, and send data procedure, while the Rx state machine does a spectrum sensing, receive RTS, send CTS, and receive data procedure. The spectrum sensing is performed using the method of averaged periodograms (Barlett's method) described in Section II. The red plot correspond to the sub-carriers whose powers are larger than a decision threshold (in this case a mean decision threshold shifted by a fixed amount of dB). One can see a DC offset due to the USRP and some occupied sub-carriers between sub-carriers 320 and 340.

Figure 2 shows one of the plots provided by the application. The different plots are the spectrum sensing and decision as shown in Figure 1 (Tab 1), the best sub-channel plot (Tab 2), the estimated channel amplitude plot (Tab 3), the power

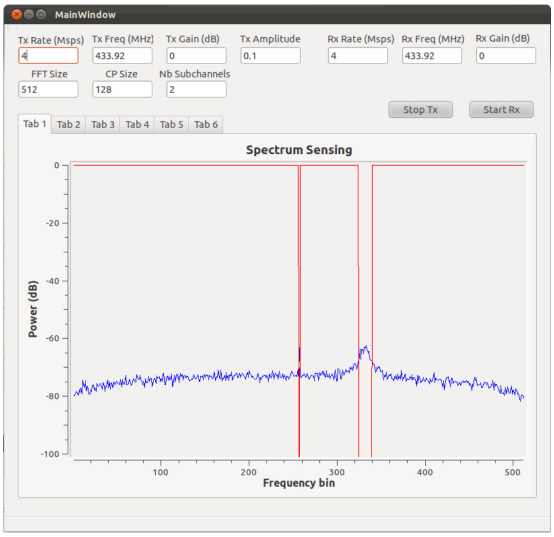

Fig. 1. Application window for spectrum sensing

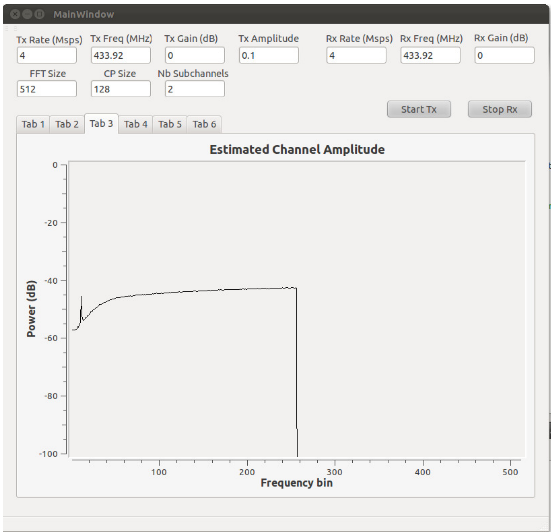

Fig. 2. Application window for channel estimation

allocation plot (Tab 4), the bit allocation plot (Tab 5). For the system to work well, a good compromise should be taken between the FFT size, CP size, the expected frequency offset, and the number of OFDM symbols used for estimation of the channel. As shown on Figure 2, a good compromise has been found with an FFT size 512, CP size 128, and number of OFDM symbols 50 such that we get a very good estimate of the channel. This serves the bit and power allocation function to have accurate values since the spectrum sensing is based on average periodograms and the estimate of the channel is based on the average of multiple OFDM symbols.

Figure 3 shows the application window for text, video and audio (Tab 6). This tab is used to input some text, video or audio in a named pipe called 'inputpipe' whenever the Tx state machine is launched. Once the enter command or the

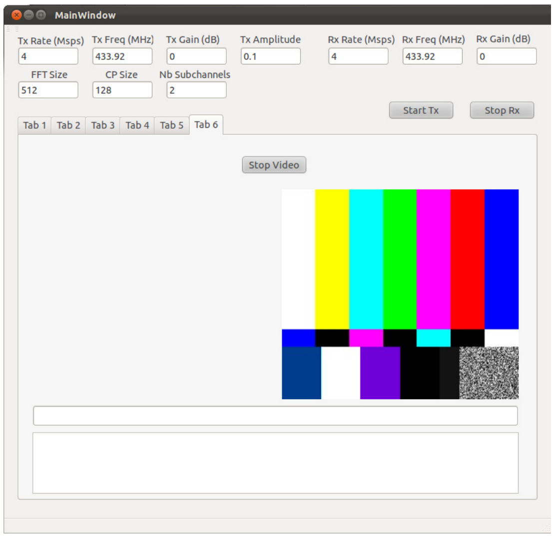

Fig. 3. Application window for text, video and audio

video button is pressed, the Tx state machine switches from a spectrum sensing state to a RTS sending state to initiate a communication. It is also used to output some text, video or audio from a named pipe called 'outputpipe' whenever the Rx state machine is launched. Once the enter command or the video button is pressed, the Rx state machine switches from a spectrum sensing state to a RTS listening state to receive a communication. After handshaking between the two CRs, data can be received as shown on Figure 3.

Future improvements of this application would be to implement other algorithms for spectrum estimates (Multitaper method, Welch, Hamming, Hanning...) and to implement decision algorithms based on noise estimation. However, a whitening approach might take a long time because of eigenvalue decomposition. Secondly, a comparison of the new OFDM signal detection algorithm with existing algorithms in the literature could also lead to an improvement of the application. Thirdly, it would be interesting to implement other algorithms for OFDM demodulation to compare different non-data-aided and data-aided approaches.

V. Conclusion

In this paper, we have proposed an adaptive OFDMA PHY/MAC on USRP platforms for a cognitive tactical radio network. In the first part of the paper, the adaptive OFDMA MAC protocol has been described, as well as the key function of the adaptive OFDMA PHY, i.e. the spectrum sensing, the OFDM signal detection, the blind OFDM demodulation and the distributed bit and power allocation. In the second part of the paper, the adaptive OFDMA PHY/MAC implementation on USRP platforms using Qt4/IT++ and the UHD API has been described. Several classes have been implemented, as well as support for text, video and audio transmission.

REFERENCES

[1] J. Mitola III and J. Gerald Q. Maguire, "Cognitive Radio: Making Software Radios More Personal," *IEEE Personal Communications*, vol. 6, no. 4, pp. 13–18, Aug. 2009.

[2] FCC, "Spectrum Policy Task Force Report," *ET Docket*, no. 02-135, 2002.

[3] O. Younis, L. Kant, A. McAuley, K. Manousakis, D. Shallcross, K. Sinkar, K. Chang, K. Young, C. Graff, and M. Patel, "Cognitive Tactical Network Models," *IEEE Communications Magazine*, vol. 48, no. 10, pp. 70–77, Oct. 2010.

[4] E. Hossain and V. Bhargava, Eds., *Cognitive Wireless Communication Networks*. Springer, 2007.

[5] W. Yu, G. Ginis, and J. M. Cioffi, "Distributed Multiuser Power Control for Digital Subscriber Lines," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 5, pp. 1105–1115, Jun. 2002.

[6] F. Wang, M. Krunz, and S. G. Cui, "Price-based spectrum management in cognitive radio networks," *IEEE Journal on Selected Areas in Communications*, vol. 2, no. 1, pp. 74–87, 2008.

[7] G. Scutari, D. P. Palomar, and S. Barbarossa, "Optimal linear precoding strategies for wideband noncooperative systems based on game theory part II: Algorithms," *IEEE Transactions on Signal Processing*, vol. 56, no. 3, pp. 1250–1267, 2008.

[8] P. Setoodeh and S. Haykin, "Robust Transmit Power Control for Cognitive Radio," *Proceedings of the IEEE*, vol. 97, no. 5, pp. 915–939, May 2009.

[9] R. H. Gohary and T. J. Willink, "Robust IWFA for open-spectrum communications," *IEEE Transactions on Signal Processing*, vol. 57, no. 12, pp. 4964–4970, Dec. 2009.

[10] M. Hong and A. Garcia, "Averaged Iterative Water-Filling Algorithm: Robustness and Convergence," *IEEE Transactions on Signal Processing*, 2011.

[11] V. Le Nir and B. Scheers, "Improved Coexistence between Multiple Cognitive Tactical Radio Networks by an Expert Rule based on Subchannel Selection," in *Wireless Innovation Forum European Conference on Communications Technologies and Software Defined Radio (SDR'11-WInnComm-Europe)*, Brussels, Belgium, Jun. 2011.

[12] ——, "Iterative Waterfilling Algorithm with Sub-channel Selection for the Coexistence of Multiple Cognitive Tactical Radio Networks," in *Military Communications and Information Systems Conference (MCC'2011)*, Amsterdam, Nederlands, Oct. 2011.

[13] J. Mo, H.-S. W. So, and J. Walrand, "Comparison of Multichannel MAC Protocols," *IEEE Transactions on Mobile Computing*, vol. 7, no. 1, pp. 50–65, Jan. 2008.

[14] Y.-E. Lin, K.-H. Liu, and H.-Y. Hsieh, "Design of Power Control Protocols for Spectrum Sharing in Cognitive Radio Networks: A Game-Theoretic Perspective," in *IEEE Conference on Communications (ICC'2010)*, Cape Town, South Africa, May 2010.

[15] L. Yang, W. Hou, L. Cao, B. Y. Zhao, and H. Zheng, "Supporting Demanding Wireless Applications with Frequency-agile Radios," in *Proceedings of the 7th USENIX Symposium on Networked Systems Design and Implementation*, April 2010.

[16] J. Blanchette and M. Summerfield, Eds., *C++ GUI Programming with Qt 4, Second Edition*. Prentice Hall, 2008.

[17] IT++ project.
http://itpp.sourceforge.net/current/index.html.

[18] "Universal Software Radio Peripheral" hardware driver.
http://code.ettus.com/redmine/ettus/projects/uhd/wiki.

[19] J. G. Proakis and D. G. Manolakis, Eds., *Digital Signal Processing. Principles, Algorithms and Applications. 3rd Edition.* Prentice Hall, 1996.

[20] V. Le Nir, T. van Waterschoot, M. Moonen, and J. Duplicy, "Blind CP-OFDM and ZP-OFDM parameter estimation in frequency selective channels," *EURASIP Journal on Wireless Communications and Networking Volume 2009, Article ID 315765, 10 pages doi:10.1155/2009/315765*, 2009.

[21] S. Chaudhari, V. Koivunen, and H. V. Poor, "Autocorrelation-Based Decentralized Sequential Detection of OFDM Signals in Cognitive Radios," *IEEE Transactions on Signal Processing*, vol. 57, no. 7, pp. 2690–2700, Jul. 2009.

[22] D. Danev, "On Signal Detection Techniques for the DVB-T Standard," in *Proceedings of the 4th International Symposium on Communications, Control and Signal Processing (ISCCSP'2010)*, Limassol, Cyprus, Mar. 2010.

[23] D. Danev, E. Axell, and E. G. Larsson, "Spectrum sensing methods for detection of DVB-T signals in AWGN and fading channels," in *Proceedings of the IEEE 21st International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC'2010*, Istanbul, Turkey, Sep. 2010.

[24] J.-J. van de Beek, M. Sandell, and P. O. Borjesson, "ML Estimation of Time and Frequency Offset in OFDM Systems," *IEEE Transactions on Signal Processing*, vol. 45, no. 7, pp. 1800–1805, Jul. 1997.

[25] H. Meyr, marc Moeneclaey, and S. A. Fechtel, Eds., *Digital Communications Receivers: Synchronization, Channel Estimation, and Signal Processing*. John Wiley and Sons, 1998.

[26] M. Rezki, L. Sabel, and I. Kale, "High Order Modulation Stripping for Use by Synchronisation Algorithms in Communication Systems," in *Proceedings of the Instrumentation and Measurement Technology Conference (IMTC'2003*, Vail, USA, May 2003.

[27] S. Boyd and L. Vandenberghe, "Convex Optimization," *Cambridge University Press*, 2004.

[28] J. M. Cioffi, "A Multicarrier Primer," *ANSI Contribution T1E1.4/91-157*, Nov. 1991.