Implementation of an Opportunistic Spectrum Access Scheme with Disruption QoS

Anirudha Sahoo, Michael Souryal, and Mudumbai Ranganathan
National Institute of Standards and Technology
100 Bureau Drive, Gaithersburg, MD, 20899, USA
email: {anirudha.sahoo, souryal, mranga}@nist.gov

***Abstract*—Opportunistic Spectrum Access (OSA) is one of the models proposed for Dynamic Spectrum Access (DSA). In OSA systems, it is important to restrict the interference caused by Secondary Users (SUs) to the Primary Users (PUs). In [1], such an OSA scheme, based on residual idle time distribution of PU traffic, was proposed for a single channel system. In this paper, we suitably modify the above scheme for OFDM-based communication. We present a prototype implementation of the scheme on a GNU Radio based software defined radio (SDR) system using USRP hardware. To the best of our knowledge, this is the first prototype implementation of an OSA system which can bound the interference to PUs. We use two different Quality of Service (QoS) metrics to validate the scheme. We also show that a naive method of transmitting for a certain fraction of the mean residual idle time may lead to poor performance.**

I. Introduction

Dynamic Spectrum Access (DSA) is being viewed as a technology to address the spectrum crunch exacerbated by static allocation. In a DSA system, spectrum is allocated to Primary Users (PU). But the spectrum can also be used by Secondary Users (SU) as long as SUs do not cause significant interference to the PUs. There have been three models proposed for DSA systems in the literature. They are *underlay*, *overlay* and *interweave* models (see [2], [3] for description of different DSA models). In this paper, we study a DSA mechanism based on interweave model. In the interweave model, SUs look for idle periods (also called white spaces) in the spectrum and then transmit in those white spaces. But the SUs have to vacate any spectrum that PUs reenter such that the interference caused to the PUs is kept low[1]. Since SUs, in this model, use the spectrum opportunistically, it is also referred to as *Opportunistic Spectrum Access* (OSA).

In the OSA model, providing Quality of Service (QoS) to PUs is important for the PU service provider. This can enable PUs to know how much performance degradation they can expect while operating in OSA mode. The QoS guarantee can also encourage service providers in legacy systems to adopt OSA technology. We refer to this QoS as *Disruption QoS*. There are few schemes proposed in the literature which provide disruption QoS. The OSA schemes presented in [4], [1] restrict the PU interference probability to below a given threshold. In the scheme proposed in [4], the SU keeps track of the amount of idle time elapsed in the current idle cycle and keeps transmitting as long as the probability of interference to the PU, conditioned on the elapsed idle time, remains below a given threshold. One shortcoming of this scheme is that an SU has to continuously sense the spectrum to keep track of the idle and busy periods. Authors in [1] proposed a scheme called Residual Idle Time Based Scheme (RIBS), in which the maximum duration of SU transmission is computed based on the residual idle time distribution and the interference probability constraint. RIBS does not require continuous sensing of the spectrum. However, this scheme was implemented in a simulation environment and for a single channel OSA system. In [5], we adapted RIBS to an OFDM based OSA system and demonstrated a preliminary implementation of RIBS over a GNU Radio Software Defined Radio (SDR) system. The implementation provides disruption QoS to PUs by constraining the interference probability to below a given threshold. However, this implementation does not fully exploit white spaces in the frequency domain. It divides the OFDM spectrum into multiple *logical channels* or *resource blocks*, but searches for white space in only one resource block at a time. So, if other resource blocks are also idle, then it misses out on those white spaces. In this paper, we address this issue by designing a method in which an SU can transmit on white spaces available on multiple resource blocks. In addition, we also propose and implement a different disruption QoS metric which provides bound in terms of probability of SU overlapping with the PU by a predetermined duration.

The work reported in this paper has three main contributions. First, this is one of the very few prototype implementations in this field. Second, although there are some research work which provides disruption QoS, they are mostly theoretical or simulation based. To the best of our knowledge, this is the first prototype implementation of a DSA system which can provide disruption QoS to the PUs. This makes the scheme more attractive from the PU service provider point of view. Third, our implementation adapts RIBS to an OFDM system with a more efficient white space access scheme compared to what we reported in [5].

II. Related Work

RIBS needs PU channel occupancy statistics. There have been quite a few prior research works reported for OSA systems which are based on channel occupancy models. Authors in [6] model the channel occupancy of a WLAN system as a 2-state semi-Markov chain. In [7], an OSA system with N PU channels is presented and modeled as 2^N-state Continuous Time Markov Chain (CTMC). In [8], [9], a Partially Observable Markov Decision Process (POMDP) based framework is proposed for an OSA system in which both PU and SU

[1]The policy governing this spectrum sharing should define “low” and “significant” interference.

CROWNCOM 2014, June 02-04, Oulu, Finland

DOI 10.4108/icst.crowncom.2014.255337

networks are assumed to be slotted. Channel selection and switching mechanisms have been studied in [10], [11] with an ON/OFF model to minimize disruptions to PUs. It uses CTMC to model channel occupancy by PUs and assumes SUs to be time-slotted which use periodic sensing. Channel utilization, in this scheme, is maximized while limiting interference to the PUs. In contrast to the above Markov based models, RIBS does not assume Markovian models for PU idle and busy time distribution, i.e., RIBS can be applied to OSA systems with any idle and busy time distribution. The impact of primary ON/OFF traffic on the capacity of a shared spectrum system is studied in [12]. The impact of variation of inter-sensing duration on the trade-off between sensing efficiency and PU interference was studied through scheduling of the spectrum sensing interval in [13], [14], [15].

Different disruption QoS metrics have been proposed for OSA systems in the literature. The scheme proposed in [16] can provide disruption QoS either in terms of collision probability or in terms of duration of interruption to an affected PU. The OSA model based on POMDP presented in [8] maximizes SU network throughput while limiting the probability of an SU colliding with PUs below a given threshold. Sung et al. [17] presented an OSA model which maximizes SU spectrum utilization while keeping the probability of interference below a given threshold. In [18] Nasreddine et al. have proposed a scheme to compute ON and OFF durations of an SU. SUs do not sense the channel; they transmit during the ON period and remain silent for the OFF period. They have used average fraction of time during which a PU is interfered by SU transmissions as a QoS metric.

III. Overview of RIBS

RIBS is an OSA scheme which provides disruption QoS to the PUs. This was first proposed by Sharma and Sahoo for a single channel PU network [1]. In this section, we present a brief overview of RIBS for single channel system. Then we explain how we have adapted RIBS to an OFDM system.

RIBS models busy and idle durations of the PUs as an *Alternating Renewal Process* (ARP) in which the process alternates between idle and busy periods. Sensing of the channel by the SU is done randomly and is modeled as a random incident into an ARP. The theory of random incidence into a renewal process is used to compute the distribution of residual idle time [19, pp. 331]. Based on the residual idle time distribution, the maximum duration, denoted as y_{max}, for which the SU can transmit is computed such that the interference to the PU is kept below a predefined threshold. RIBS assumes that channel occupancy, i.e., the busy and idle time distribution, is known and does not change for a long time. So, the maximum duration y_{max} is computed based on the channel occupancy and is valid as long as the channel occupancy parameters do not change. Note that RIBS does not assume any particular distribution (e.g., exponential distribution).

Channel access by an SU, shown in Figure 1, is quite simple. If the SU senses the channel to be idle, then it transmits for $(y_{max} - s)$ duration, where s is the sensing duration (ab in the figure). After its transmission is over (point c), it generates an exponentially distributed backoff value (ad) to determine the next sensing point. The backoff is applied with respect to the previous sensing point (a). It is possible that the backoff value may not be long enough to go beyond the current time (point c). In such cases, multiple backoff values are generated until their cumulative value goes beyond the current time. If the channel is found to be busy then the SU goes into backoff (as shown by point d in the figure).

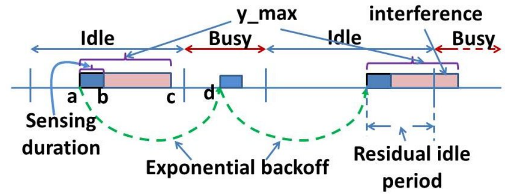

Fig. 1. Illustration of SU Channel Access in RIBS

IV. Adaptation of RIBS to an OFDM System

The work reported in [1] simulated RIBS for a single channel system. In this paper, we present the methodology by which RIBS can be applied to an OFDM system and implement it using GNU Radio on USRP[2].

In an OFDM system, spectrum is divided into multiple sub-carriers. We assume that a sender uses sets of contiguous sub-carriers to transmit data. We refer to a set of contiguous sub-carriers used by a sender as a *resource block*. The available sub-carriers are divided into resource blocks and nodes communicate using one or more of these resource blocks. Each resource block is represented by a *carrier map*. A carrier map is a bitmap consisting of a set of 0's and 1's and its length is equal to the number of available sub-carriers in the system. A carrier map corresponding to a resource block has contiguous 1's at the bit positions corresponding to the contiguous sub-carriers used by the resource block. For example, if the total number of available sub-carriers is 240 and a resource block uses sub-carriers 1 through 60, then its carrier map will be of length 240 and will have sixty 1's from bit position one to sixty and 0's in all other bit positions.

Each resource block has idle and busy periods which are modeled as an ARP. The idle and busy periods are determined by the traffic pattern transmitted by PUs on the resource blocks. Let I^i and RI^i be the random variables representing idle time and residual idle time duration of a resource block B^i. Sensing instants are treated as random incidences into the ARP. Thus, spectrum access of an SU can be modeled as random incidence into an ARP. From the theory of random incidence into a renewal process we have [19, pp. 331]:

$$f^i_{RI}(y) = \frac{1 - F^i_I(y)}{E[I^i]} \tag{1}$$

$$F^i_{RI}(y) = \int_0^y f^i_{RI}(z)dz \tag{2}$$

where $f^i(.)$ and $F^i(.)$ represent the pdf and cdf, respectively, and $E[I^i]$ denotes the expected idle time of resource block B^i.

The above theory is used to provide disruption QoS to the PUs by the SUs by conforming to the constraints based on the

[2]The identification of any commercial product or trade name does not imply endorsement or recommendation by the National Institute of Standards and Technology, nor is it intended to imply that the materials or equipment identified are necessarily the best available for the purpose.

disruption QoS metric of the OSA system, as explained in the following sections.

A. Disruption QOS Metric: Probability of Interference

Probability of interference (PoI) is defined as the probability that an SU transmission interferes with the PUs. It is a natural metric to measure disruption to PU and has been used in the literature [1], [4]. In this mode, a PoI threshold or constraint (η^i) for the resource block B^i is specified to the SU and the SU controls its transmission duration y such that its probability to interfere with PUs remains below the specified PoI constraint. Note that an SU transmitting over resource block B^i interferes with the PUs when $RI^i < y$. Thus, the interference probability constraint is given by

$$F_{RI}^i(y) \leq \eta^i \tag{3}$$

The maximum duration an SU can transmit on resource block B^i, is given by the maximum value of y (say y_{max}^i) which satisfies (3).

In this paper, we study an OFDM system in which the idle and busy periods on each resource block are exponentially distributed. It has been shown in [19, pp. 331] that if $F_I^i(y)$ is exponentially distributed, then $F_{RI}^i(y)$ is also exponentially distributed with the same parameter and is given by

$$F_{RI}^i(y) \;=\; 1 - e^{-\lambda_I^i y} \tag{4}$$

where $1/\lambda_I^i$ is the mean idle time of PU traffic on resource block B^i. So, when the PU idle time duration is exponentially distributed, the interference probability constraint is given by (by combining (3) and (4))

$$e^{-\lambda_I^i y} \;\geq\; 1 - \eta^i \tag{5}$$

Since $e^{-\lambda_I^i y}$ is a decreasing function of y, the inequality in (5) is replaced with equality to get the maximum value of y (y_{max}^i) which satisfies the PoI constraint.

B. Disruption QOS Metric: Overlap Threshold Probability

For some applications, the probability of interference may not be an appropriate QoS. A more appropriate disruption QoS metric may be one which can restrict the co-channel overlap between SU and PU transmissions. So, we define *overlap threshold probability* (OTP) as the probability that, given an interference event, the overlap of SU transmission with PU transmission goes above a predefined *overlap threshold duration* (OTD). So, an OSA system which implements this metric has to define two parameters: an OTP constraint and an OTD for which the OTP constraint has to be met. Obviously, the lower the OTP and OTD, the better is the QoS provided to the PUs.

Let T_{th}^i be the overlap threshold duration and γ^i be the OTP constraint for resource block B^i. If SU transmits for duration y over resource block B^i, then as per definition of OTP, we have

$$\begin{aligned} P[(y - RI^i) > T_{th}^i | (RI^i < y)] &\leq \gamma^i \\ P[RI^i < (y - T_{th}^i) | (RI^i < y)] &\leq \gamma^i \\ \frac{P[RI^i < (y - T_{th}^i) \; and \; RI^i < y]}{P[RI^i < y]} &\leq \gamma^i \\ \frac{F_{RI}^i(y - T_{th}^i)}{F_{RI}^i(y)} &\leq \gamma^i \end{aligned} \tag{6}$$

When the idle time duration of PU is exponentially distributed, using (6), we have

$$\frac{1 - e^{-\lambda_I^i (y - T_{th}^i)}}{1 - e^{-\lambda_I^i \cdot y}} \;\leq\; \gamma^i \tag{7}$$

After rearranging the terms we get

$$e^{-\lambda_I^i \cdot y} \;\geq\; \frac{1 - \gamma^i}{e^{\lambda_I^i \cdot T_{th}^i} - \gamma^i} \tag{8}$$

The maximum value of y for resource block B^i (y_{max}^i) is calculated with the inequality in (8) replaced by equality.

C. Channel Access by SU in an OFDM system

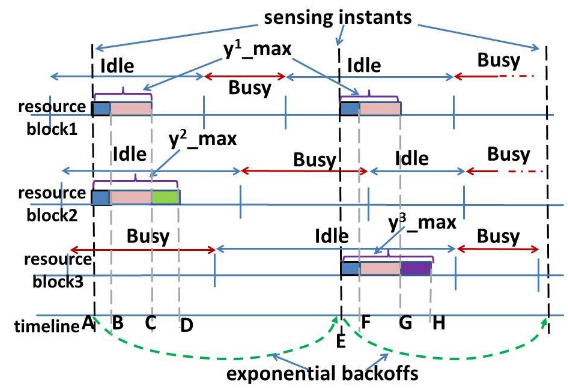

Fig. 2. Illustration of SU Channel Access in an OFDM System using RIBS

In this implementation, a common sensing schedule for all the resource blocks is followed, i.e., an SU senses all the resource blocks at a time and transmits over all the resource blocks which were found idle. This is in contrast to our earlier implementation reported in [5], in which an SU senses one resource block at a time and transmits on that resource block if it is found idle. Thus, if any other resource blocks were idle at that time, the SU misses out on those white spaces. So, the current implementation is more efficient in terms of white space usage. The transmission over the idle resource blocks is carefully designed such that it transmits for the maximum duration allowed (based on the computation as per the disruption QoS metric used) on the respective resource blocks. We explain the channel access with the help of Figure 2. We show the channel occupancy of three resource blocks. Let us assume that the maximum duration for which SU is allowed to transmit on these resource blocks are y_{max}^1, y_{max}^2 and y_{max}^3 respectively. Assume that $y_{max}^1 < y_{max}^2 < y_{max}^3$. At time instant A, the SU starts sensing for a duration $s = AB$. It finds that resource blocks 1 and 2 are idle. So, it transmits on both the resource blocks for duration $(y_{max}^1 - s)$ until time instant C. Note that SU would use a carrier map which is the bit-wise OR of the carrier maps of resource blocks 1 and 2 to transmit on those two resource blocks. At C, it changes its carrier map to the carrier map assigned to resource block 2 and transmits for duration $(y_{max}^2 - y_{max}^1)$ until time D. Thus, it effectively transmits over resource block 1 and 2 for $(y_{max}^1 - s)$ and $(y_{max}^2 - s)$ duration respectively. We refer to each transmission as a *transmission opportunity*. At time instant D, the SU computes the next sensing instant by generating a backoff which is exponentially distributed and applies it with respect to the previous sensing instant A. The SU then waits until next sensing instant (E) to sense the spectrum.

V. Details of Experiments

A. Testbed Setup

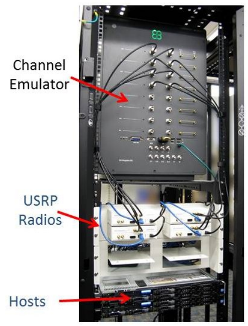

Fig. 3. Testbed used for implementation of RIBS

To evaluate the performance of RIBS, we have set up a testbed shown in Figure 3. The testbed consists of two rack-mounted servers (or hosts), each having 12 core CPUs and 64 GB memory. Both the servers run CentOS linux 2.6.32. Four USRP N210s having SBX daughterboards, which can operate between 400 MHz to 4400 MHz, are mounted on the same rack. Each server drives two USRPs. The four USRPs are connected in a full mesh topology through a channel emulator. Having USRPs communicate through a channel emulator has many advantages. It is easy to create different channel models with the channel emulator. Since the USRPs do not radiate into the air, it can be operated in licensed spectrum without having a license. It also enables repeatability of the experiments with the exact same channel condition.

B. Configuration of Experiments

For our experiments, we designated two USRPs as a PU sender and receiver pair whereas the other two USRPs acted as a SU sender and receiver pair. The center frequency of the carrier was set at 795.5 MHz. The total OFDM bandwidth was 1MHz. The bandwidth was divided into 512 sub-carriers, but only the middle 240 sub-carriers were used for data transmission. So the last 136 sub-carriers on either side of the center frequency were left out since they may not be suitable for data transmission due to tapering effect of the low pass filter applied to the base band signal. The 240 sub-carriers were divided into four resource blocks of 60 sub-carriers each. The lowest resource block was reserved for control messages and the other three resource blocks were used for data transmission. The PU sender transmitted over the data resource blocks in such a way that the traffic idle and busy periods on the individual resource blocks were exponentially distributed. The PU and SU sender transmitted with power densities of -92.2 dBm/Hz and -97.7 dBm/Hz respectively using BPSK modulation with a cyclic prefix length of 128 samples. Line of sight propagation with fixed path loss of 22 dB was used for all the links in the mesh. No channel coding was used. The PU and SU senders were assumed to have saturated traffic, i.e., they always had a packet to transmit. All the parameters used in our experiments are given in Table I.

	tx power density (dBm/Hz)	band width (MHz)	mean idle/busy period (s)	carrier freq (MHz)	packet size (bytes)
PU	-92.2	1	10/10 5/5 4/4	795.5	50
SU	-97.7	1	N/A	795.5	50

TABLE I
Parameter Values used in the experiments

C. Implementation of RIBS on the Testbed

To implement RIBS, we started with a software platform called *Papyrus* [20]. Papyrus implemented full duplex communication using the GNU Radio OFDM implementation over USRP. However, Papyrus was developed on an older version of GNU Radio and an older version of USRP hardware (USRP1). We ported it to a recent version of GNU Radio (version 3.6.5) and latest USRP hardware (USRP N210). In our OSA system, the SU has to tell the receiver which carrier map (corresponding to the available resource blocks) it is going to use for transmission. The SU receiver then sends an ACK and tunes to that carrier map. After receiving the ACK, the SU sender starts transmitting data over those resource blocks. Although Papyrus had the above basic *handshake* mechanism in place, we discovered that it was not very robust due to some state machine and thread synchronization issues and due to many *busy waits* in the code. We fixed those issues in Papyrus and then implemented the RIBS channel access scheme presented in Section IV-C.

VI. Results

A. Probability of Interference

The first set of experiments we ran was to verify that RIBS does provide disruption QoS in terms of PoI. Experiments were carried out with three different values of the mean backoff value (BO), as shown in Table II. For each BO value, the PoI constraint (η) was set to 0.1 and 0.2 and various performance measurements were taken for each PoI setting. We also experimented with a naive SU access scheme, denoted as *RI_0.5*, in which the SU transmits for half the mean residual idle time whenever it senses resource blocks to be idle. This scheme is used to show that a naive scheme which transmits only for a fraction of mean residual idle time, although intuitively might appear to be fine, does not perform well. In terms of performance metrics for the PUs, we measured the PoI and *packet loss rate*. Packet loss rate is the number of packets, expressed as percentage of the total packets sent by the PU sender, which could not be received by the PU receiver application. For the SU system, we measured SU throughput and the number of sensing events generated by the SU sender. Since sensing consumes energy, the number of sensing events is an indicator of sensing overhead (or the amount of energy consumption) in achieving the corresponding throughput.

The first configuration was with η set to 0, i.e., there was no SU transmission, to provide the baseline performance of the PU. We see a rather large packet loss in this configuration. We traced the cause of this to two factors. First, there was a problem in the basic implementation of OFDM in GNU Radio (*benchmark_tx.py*), on which the Papryrus implementation is based. The GNU Radio OFDM implementation is meant for streaming data. When it is used to send bursty data, it loses a few packets at the beginning of each burst. We are currently investigating the problem. Second, we have not used any channel coding. So, even a single bit error in a packet leads to a packet loss.

In the second set of configurations, the mean backoff value was set to $\min(y^i_{max})$, which is the minimum of the y_{max} values across the three resource blocks. Since y^i_{max} of resource block B^i is a function of η (see Eqn 3), the BO value increases as the PoI constraint increases. For $\eta = 0.1$, the y_{max} values for the three resource blocks were [1.05, 0.53, 0.42]s, whereas for $\eta = 0.2$, the y_{max} values were [2.23, 1.12, 0.89]s. For RI_0.5 scheme the y_{max} values were [5, 2.5, 2.0]s. So the y_{max} values of RI_0.5 scheme were the highest followed by RIBS with $\eta = 0.2$, followed by RIBS with $\eta = 0.1$. For these three configurations, the BO values were 0.42s, 0.89s and 2.0s respectively. We observe that as η increases, y_{max} of the resource blocks increase, which leads to more interference from the SU. Hence, this leads to higher PU packet loss. Compared to the baseline case ($\eta = 0.0$), the PU packet loss rate is quite high. We traced this to the fact that, although the PoI constraint was satisfied, the durations of overlap of some interfering SU transmissions were long. So, we are currently looking into devising an OSA scheme which will constrain the overall duration of overlap. For the $RI\_0.5$ scheme, the packet loss is even higher, because, in this case, the values of y_{max} are more than the two RIBS configurations, which leads to more interference. In terms of SU performance, we see that the SU throughput more or less remains the same, even though y_max increases (as we go from RIBS with η equal to 0.1 and 0.2 to RI_0.5 scheme). Intuitively, one would expect SU throughput to increase as the PoI constraint is relaxed. However, in this set of configurations, BO is set to the minimum of the y_{max} values. Thus, as y_max increases, so does the backoff value. An increase in backoff value leads to SU skipping more white spaces for transmission, which translates to fewer transmission opportunities. So, the increase in throughput due to increase in y_max value is countered by the decrease in number of transmission opportunities. But the sensing overhead decreases with increase in η and with RI_0.5 scheme due to the increase in BO value.

In the third set of configurations, the mean back off value was set to $(0.1) \cdot \min(cycle^i)$, which is 10% of the minimum of mean ARP cycles across the three resource blocks. An ARP cycle is the duration of a busy period and the next idle period. The mean ARP cycle of the three resource blocks is $[20, 10, 8]$s. Thus, the BO is set to 0.8s in this set of configurations. Note that, unlike the previous set of configurations, the BO value remains the same as η changes or as we run the RI_0.5 scheme. The PU packet loss rate has the same trend as the previous set of configurations. SU throughput increases as η increases and as we follow the RI_0.5 scheme. This is expected, because the SU transmission duration increases as η increases, while BO remains constant. Since y_{max} for the RI_0.5 scheme is the highest, this scheme results in highest SU throughput. The sensing overhead decreases as η increases and is the lowest for RI_0.5 scheme. Since BO does not change, the backoff incident points are the same across all configurations in this set. But, when the transmission duration increases (due to increase in η or due to the RI_0.5 scheme), some of the backoff points fall inside the transmission duration and hence are not used for sensing. Thus, the number of sensing instances decreases.

In the fourth set of configurations, the mean back off value was set to $(0.05) \cdot \min(cycle^i)$, which is 5% of the minimum mean of ARP cycles across the three resource blocks. So, the BO is half of the BO in the previous set of configurations or 0.4s. The results across configurations within this set are very similar to the previous set. When compared with the respective configurations in the previous set, we see that the SU throughput is higher because the BO is now lower than in the previous set which leads to more transmission opportunities. More transmission opportunities implies higher sensing overhead and more interference to the PU. Hence, PU packet loss rate is higher for these configurations compared to the previous set.

In all sets of the configurations, we see that with RIBS, the measured PoI is marginally higher than the corresponding η value. This violation is due to deviations of implementation from ideal RIBS. The latency from the instant when a packet send command is issued from the application to the time instant when the USRP actually sends the packet is not accounted for. Furthermore, the PU traffic is not perfectly exponentially distributed because of a 50ms forced delay we had to introduce to give the PU receiver time to reset its carrier map when requested by the PU sender. The measured PoI and the PU packet loss rate for the RI_0.5 scheme is much worse than that of RIBS. This shows that a naive scheme would not perform well in an OSA system.

B. *Overlap Threshold Probability*

The set of configurations for this QoS metric was very similar to that of the PoI metric. The main difference was in the calculation of y_{max}, which was calculated such that inequality (8) holds. In the set of experiments presented under this QoS metric, two overlap threshold durations were used: 0.1 and 0.2 of mean PU busy time. The RI_0.5 scheme remained the same as in the previous set of experiments. However, to measure OTP for the RI_0.5 scheme, we split it into two categories. The first one is denoted as RI_0.5_ov0.1, in which we examined the RI_0.5 SU transmissions which overlapped with the PU transmission by more than 0.1 of the mean busy time. The second one, denoted as RI_0.5_0.2, looked for SU transmissions which overlapped with PU transmissions by more than 0.2 of the mean busy time. Then OTP was computed by dividing the number of overlapped transmissions which exceeded the threshold duration by the total SU transmissions which overlapped with the PU. For all the configurations, the OTP constraint (γ^i) was set to 0.05, i.e., no more than 5% of all the interference events exceeded the overlap threshold. The results for experiments with different sets of configuration are shown in Table III.

As the OTP constraint increases, we see a trend very similar to what we saw in the previous set of configurations involving the PoI QoS metric, with PU packet loss rate, SU throughput and sensing overhead. We see that the measured OTP is below the threshold in most cases. Again, we see that the naive RI_0.5_ov0.1 and RI_0.5_ov0.2 schemes perform much worse than RIBS in terms of OTP QoS metric and PU packet loss rate. For both the metrics, the confidence interval (with 95% confidence) for packet loss rate is 0.1% or less for all the sets

mean backoff	PoI constraint (η)	measured PoI	PU packet loss rate (%)	SU throughput (kb/s)	number of sensing events
	0.0	N/A	12.16	N/A	N/A
BO=$\min(y^i_{max})$	0.1	0.113	24.96	23.63	1905
	0.2	0.205	26.24	24.05	988
	RI0.5	0.368	29.05	23.4	482
BO=$0.1 \cdot \min(cycle^i)$	0.1	0.113	21.04	16.4	1459
	0.2	0.209	26.93	25.03	1188
	RI0.5	0.394	36.47	32.25	714
BO=$0.05 \cdot \min(cycle^i)$	0.1	0.108	23.90	23.14	2057
	0.2	0.218	31.40	31.43	1540
	RI0.5	0.385	41.67	37.65	847

TABLE II
EXPERIMENTAL RESULTS FOR DISRUPTION QOS METRIC POI

mean backoff	overlap threshold duration (fraction of mean busy time)	measured OTP	PU packet loss rate (%)	SU throughput (kb/s)	number of sensing events
	0.0	N/A	12.16	N/A	N/A
BO=$\min(y^i_{max})$	0.1	0.049	24.15	22.49	2027
	0.2	0.052	25.89	23.96	1199
	RI0.5_ov0.1	0.252	29.05	23.4	482
	RI0.5_ov0.2	0.16	29.05	23.4	482
BO=$0.1 \cdot \min(cycle^i)$	0.1	0.048	20.80	16.53	1475
	0.2	0.046	25.98	24.32	1213
	RI0.5_ov0.1	0.26	36.47	32.25	714
	RI0.5_ov0.2	0.16	36.47	32.25	714
BO=$0.05 \cdot \min(cycle^i)$	0.1	0.048	24.43	22.93	2055
	0.2	0.046	30.25	31.34	1626
	RI0.5_ov0.1	0.26	41.67	37.65	847
	RI0.5_ov0.2	0.18	41.67	37.65	847

TABLE III
EXPERIMENTAL RESULTS FOR DISRUPTION QOS METRIC OTP
($\gamma^i = 0.05$)

of configurations.

We conclude this section with some implications of these results. In an OSA system where the disruption QoS constraint can change (e.g., based on time of the day), the SU should use a mean backoff value based on the cycle time of the PU traffic (rather than based on y_{max}) so that that it can increase its throughput when the constraint is relaxed. Also, for a given QoS constraint value, as the mean backoff value decreases, the SU throughput increases, but at the cost of an increase in the sensing overhead. The SU provider should appropriately set the operating mean backoff parameters based on what throughput it wants and how much sensing overhead (energy consumption) it is willing to incur.

VII. CONCLUSION AND FUTURE WORK

In this paper, we presented a prototype implementation of RIBS [1] for OFDM based wireless communication. RIBS is based on random incidence into an alternating renewal process. The spectrum access method for the SU is very simple yet very effective in terms of providing disruption QoS. Through our experiments presented in this paper, we have shown that it is possible to limit disruption to PUs using RIBS and that naive methods such as transmitting for a certain fraction of mean residual idle time may lead to poor performance.

The current implementation is demonstrated with synthetic PU traffic and a simplistic channel propagation model. Next, we are going to work on implementing RIBS with realistic PU traffic and time-varying channel propagation models. RIBS requires the knowledge of the idle and busy time distributions of PU traffic. So, to implement RIBS in a realistic PU traffic scenario, it would require collecting and analyzing PU traffic (offline) for a sufficiently long time and fitting distributions. If the PU traffic pattern changes frequently, then the PU traffic analysis may need to be done frequently. As a result, we are looking at schemes that would estimate PU traffic parameters dynamically (on the fly) from the sensing results gathered.

REFERENCES

[1] M. Sharma and A. Sahoo, "Stochastic Model Based Opportunistic Channel Access in Dynamic Spectrum Access networks," *(To appear) IEEE Transactions on Mobile Computing*.

[2] M. Song, C. Xin, Y. Zhao and X. Cheng, "Dynamic Spectrum Access: From Cognitive Radio to Network Radio," *IEEE Wireless Communication*, vol. 19, no. 1, pp. 23–29., February 2012.

[3] Q. Zhao, and B. Sadler, "A Survey of Dynamic Spectrum Access," *IEEE Signal Processing Magazine*, vol. 24, no. 3, pp. 79–89., May 2007.

[4] A. Plummer Jr., M. Taghizadeh and S. Biswas, "Measurement Based Capacity Scavenging via Whitespace Modeling in Wireless Networks," in *IEEE GlobeComm*, pp. 1–7., December, 2009.

[5] A. Sahoo, M. Souryal and M. Ranganathan, "Providing Disruption QoS in an OFDM System Using Residual Idle Time Based Opportunistic Spectrum Access," in *IEEE Global Conference on Signals and Information Processing (GlobalSIP)*, December 2013.

[6] P. K. Tang, Y. H. Chew and L. C. Ong, "On the Distribution of Opportunity Time for the Secondary Usage of Spectrum," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 3, March 2009.

[7] S. Geirhofer, L. Tong, and B. M. Sadler, "Cognitive Medium Access: Constraining Interefence Based on Experimental Models," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 1, January 2008.

[8] Q. Zhao, L. Tong, A. Swami and Y. Chen, "Decentralized Cognitive MAC for Opportunistic Spectrum Access in Ad Hoc Networks: A POMDP Framework," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 2, pp. 589–600, April 2007.

[9] A. T. Hoang, et al., "Opportunistic Spectrum Access for Energy-constrained Cognitive Radios," *IEEE Transactions on Wireless Communications*, vol. 8, no. 3, March 2009.

[10] Q. Zhao, S. Geirhofer, L. Tong, and B. M. Sadler, "Opportunistic Spectrum Access via Periodic Channel Sensing," *IEEE Transactions on Signal Processing*, vol. 56, no. 2, pp. 785–796, February 2008.

[11] L. Yang, L. Cao, and H. Zheng, "Proactive Channel Access in Dynamic Spectrum Networks," *Physical Communications*, vol. 1, no. 2, pp. 103–111, 2008.

[12] M. G. Khoshkholgh, K. Navaie, and H. Yanikomeroglu, "On the Impact of the Primary Network Activity on the Achievable Capacity of Spectrum Sharing over Fading Channels," *IEEE Trans. Wireless Communications*, vol. 8, no. 4, pp. 2100–2111, April 2009.

[13] W. Y. Lee, and I. F. Akylidiz, "Optimal Spectrum Sensing Framework for Cognitive Radio Networks," *IEEE Trans. Wireless Communications*, vol. 7, no. 10, pp. 3845–3857, October 2008.

[14] H. Kim, and K. Shin, "Efficient Discovery of Spectrum Opportunities with MAC-layer Sensing in Cognitive Radio Networks," *IEEE Trans. on Mobile Computing*, vol. 7, no. 5, pp. 533–545, May 2008.

[15] X. Zhou et. al., "Probability based Optimization of Inter-sensing Duration and Power Control in Coginitive Radio," *IEEE Trans. Wireless Communications*, vol. 8, no. 10, pp. 4922–4927, October 2009.

[16] S. Huang, X. Liu and Z. Ding, "Opportunistic Spectrum Access in Cognitive Radio Networks," in *IEEE Infocom*, pp. 2101–2109., April, 2008.

[17] K. W. Sung, S. Kim and J. Zander, "Temporal Spectrum Sharing Based on Primary User Activity Prediction," *IEEE Transactions on Wireless Communications*, vol. 9, no. 12, pp. 3848–3855, Decemebr 2010.

[18] J. Nasreddine, J. Riihijarvi, X. Li and P. Mahonen, "Exploring Opportunisitic Access Techniques Using Stochastic Models: Dynamic Spectrum Access Without Sensing," in *IEEE Military Communications Conference*, pp. 1053–1060., November 2011.

[19] K. S. Trivedi, *Probability and Statistics with Reliability, Queuing and Computer Science Applications*. John Wiley and Sons, 2nd ed., 2002.

[20] L. Yang, Z. Zhang, W. Hou, B. Y. Zhao and H. Zheng, "Papyrus: A Software Platform for Distributed Dynamic Spectrum Sharing Using SDRs," *ACM SIGCOMM Computer Communication Review*, vol. 41, no. 1, pp. 32–37., January 2011.