On Time-domain Coexistence of Unlicensed and Licensed Spectrum Users

Ashwini Kumar, Kang G. Shin
The University of Michigan
Ann Arbor, MI 48109-2121, U.S.A.
{ashwinik,kgshin}@umich.edu

Young-June Choi
Ajou University
Korea
choiyj@ajou.ac.kr

Dragos Niculescu
University Politehnica of Bucharest
Romania
dragos.niculescu@elcom.pub.ro

***Abstract*—This paper aims to exploit spectrum white spaces in time-domain via the Dynamic Spectrum Access (DSA) technology. DSA relies on opportunistic access of licensed spectrum by unlicensed devices. Any "unlicensed-with-licensed" coexistence must ensure "safe" (i.e., un-interfered) communications for the incumbents, while achieving high spectrum-use efficiency for the secondary users. We propose a novel and comprehensive metric called the *Coexistence Goodness Factor* (CGF) to accurately model the inherent tradeoff between incumbent safety and unlicensed access efficiency for time-domain DSA-based coexistence. To optimize the coexistence performance, we propose a generic, online, dual-mode DSA coexistence protocol. The unlicensed devices attempt to estimate incumbent behavior patterns, and enter the *Aggressive Mode* (AM) once such a pattern is found, while they stay in the *Safe Mode* (SM) otherwise. For low-overhead and reliable estimation of incumbent spectrum-usage patterns, we propose algorithms based on Approximate Entropy (ApEn). Further, we design *Spectrum-Conscious WiFi* (SpeCWiFi), which provisions the proposed DSA coexistence scheme to the base 802.11 MAC. We conduct an extensive experimental evaluation of SpeCWiFi using a MadWifi-based prototype implementation in conjunction with 802.11 wireless cards. The evaluation shows that SpeCWiFi achieves excellent time-domain DSA coexistence in the presence of different types of licensed spectrum, including fast-varying channels that feature short-duration white spaces.**

I. Introduction

Dynamic Spectrum Access (DSA) [1], enabled by software-defined or cognitive radio technology [2], refers to the unlicensed wireless access paradigm in which unlicensed secondary users (SUs) can opportunistically communicate on a channel licensed to other authorized primary users (PUs). The goal of DSA is to exploit spectrum white spaces [3]. The major emphasis of DSA so far has been in frequency-domain aimed at the TV spectrum space [4], owing to recent FCC's decision to permit unlicensed access in TV channels [5].

We postulate that the advantages and scope of DSA can be significantly enhanced by effective harnessing of spectrum white spaces in time-domain. We view a time-domain approach as complementary to frequency-domain methods in providing a complete unlicensed access solution.

DSA coexistence problem: This paper explores the problem of *DSA coexistence*, defined as *time-domain unlicensed coexistence* of PUs with SUs. By this definition, "DSA coexistence" comes into play only when the channel[1] has already been selected by the SUs and should then be utilized, i.e., after the frequency-domain aspect of unlicensed access has been established through a spectrum-management component [4].

Inadequacy of state-of-art DSA coexistence: Existing DSA research lacks sufficient coverage of DSA coexistence since its primary focus has exclusively been on TV bands. For TV spectrum, DSA coexistence constitutes halting unlicensed transmissions within a specific time-frame when a PU activity is detected—IEEE 802.22 standard draft [6] provides 2 seconds for ceasing SU transmission based on FCC guidelines [7]. In strict terms, there is no genuine time-domain coexistence as SUs simply try to find and migrate to a different channel when such an event occurs.

This form of DSA coexistence, however, is ineffective in most of licensed spectrum because of the underlying diversity in incumbent channel-access characteristics. For example, the WiMAX-based IEEE 802.16h draft [8] explicitly proposes spectrum white spaces of ∼10ms duration, depending on traffic load. Such small but abundant time-domain white spaces are also seen in several spectrum bands like satellite and cellular service channels [9]. In channels characterized by *fast-varying* incumbent behavior, PUs access channels very frequently, but each access lasts for a short time, resulting in very small ON/OFF durations. Even though each PU–SU transmission overlap is controlled to be within the specified regulatory time-limit, the accumulated interference to PUs over a long term can be significant, and hence, unacceptable to licensees.

Summary of proposed approach: This work is to advance the state-of-art DSA coexistence[2] by a comprehensive exploration of the feasibility of exploiting white spaces in time-domain. We first formalize the DSA coexistence problem by taking into account two key requirements:

1) *PU-safety:* PUs must be protected from the interference generated by unlicensed SUs;
2) *SU-efficiency:* SUs should be able to maximize the utilization of available white spaces.

PU-safety and SU-efficiency inherently conflict with each other, because increase in channel utilization by SUs leads to an increase in interference to PUs. This tradeoff is more pronounced in fast-varying channels. To precisely quantify this tradeoff, we define *Coexistence Goodness Factor* (CGF). CGF

[1] We use the word "channel" to denote any portion of a licensed spectrum region. It may consist of one or more physical channels.

[2] Throughout this paper, by "DSA coexistence" or "coexistence," we mean time-domain PU–SU coexistence, unless mentioned otherwise.

is a comprehensive metric to evaluate the effectiveness of DSA coexistence protocols in any licensed spectrum.

In order to quantify the run-time performance of DSA coexistence, we define a CGF-based multi-objective function. For practical optimization of the objective function, we propose an intelligent dual-mode DSA coexistence protocol, with *Safe Mode* (SM) and *Aggressive Mode* (AM). The proposed scheme features joint transmission-cum-sensing scheduling based on *Approximate Entropy* (ApEn) [10] estimate of PUs' channel-access pattern. We also develop a proof-of-concept prototype on 802.11 MAC, called *Spectrum-Conscious WiFi* (or SpeCWiFi for short). SpeCWiFi is evaluated on a testbed in our department, by developing an implementation based on MadWifi device driver [11].

Contributions: This paper makes three main contributions. First, we identify the key challenges in achieving generic DSA coexistence and provide an objective function for its optimization in terms of CGF. To the best of our knowledge, this is the first attempt to formalize the problem of general time-domain exploitation of spectrum white spaces. Second, we provide a low-overhead and online solution for the formulated DSA coexistence problem, together with the framework and algorithms. The main advantage of the proposed approach lies in its practicality—a software-based prototype implementation illustrates its deployment feasibility. It can be incrementally applied to any DSA protocol, complementing other efforts in the field of DSA. Third, we demonstrate effective DSA coexistence using our proposed solution in testbed experiments.

Organization: The paper is organized as follows. Prior related work is discussed in Section II. We present the system model and motivation in Sections III and IV, respectively, and formalize the DSA coexistence problem in Section V. The proposed coexistence mechanisms are presented in Sections VI and VII with implementation details in Section VIII. Evaluation of SpeCWiFi is described in Section IX. The paper concludes with Section X.

II. Related Work

There is a large body of research related to the development of DSA; see [1] for a general survey.

Spectrum surveys and analysis: Several spectrum surveys and studies have been conducted in various places (both urban and rural) around the globe [3], [9], which confirm the existence of vast spectrum white spaces in licensed spectrum, highlighting the potential of DSA. Further, the survey in [9] indicated that spectrum white space characteristics in channels exhibit significant correlation, including temporal correlation. It provides a basis for developing methods based on pattern estimation and prediction to exploit such spectrum opportunities.

Platforms and standards: Use of cognitive radios for DSA has been discussed in [2], [12], [13]. Experimental software-defined radios for testing DSA protocols have been explored in [14], [15]. On the administrative front, the US FCC has approved its preliminary guidelines for unlicensed spectrum usage in TV bands [5], [7]. The IEEE 802.22 Group [6] is standardizing WRANs in TV bands.

Spectrum sensing: Well-known techniques for spectrum sensing are matched-filter coherent detection, energy detection, and cyclostationary feature detection [16]. To enhance the effectiveness of physical-layer techniques, higher-layer spectrum sensing mechanisms, especially sensing scheduling, have also been investigated [17], [18]. Any reliable spectrum-sensing scheme can be applied to our DSA coexistence approach.

PU-pattern estimation: Some MAC-layer sensing scheduling schemes estimate the incumbent's channel-usage pattern. Our estimation based on Approximate Entropy (ApEn) [10] differs from such existing techniques in incurring low-overhead and being generic. For example, it makes no restrictive assumptions like i.i.d. ON and OFF durations for PUs as in [18].

While [9] proposes a 2-D frequent pattern mining algorithm to predict channel availability, it cannot be used online as it entails significant computational overhead and requires a long duration of training data (e.g., at least 2 hours).

An experimental study of the usefulness of ApEn metric in the context of DSA was presented in [19]. However, its focus was on improving the reliability of spectrum sensing rather than DSA coexistence.

DSA MAC protocols: Several DSA MAC protocols have been proposed in the research community [20], [21]. Application-awareness framework for DSA MAC design has been proposed in [22] highlighting the benefits of a cross-layer approach with DSA MAC. Cross-layer with DSA MAC has also been exploited to improve TCP performance in [23]. Instead of proposing a new DSA MAC, we modify the existing 802.11 MAC to design and implement our DSA coexistence scheme. In this regard, our approach is similar to the WhiteFi MAC protocol [4] that aims to achieve WiFi-like networking in the UHF spectrum space with DSA.

Self-coexistence: DSA coexistence refers to spectrum sharing between any SU and incumbents, while self-coexistence refers to sharing of available spectrum white spaces between competing SUs. In licensed channels, self-coexistence can be considered as complementary to DSA coexistence. Self-coexistence in DSA has been extensively studied in Sparta [24] and Flex [25] proposals. Both [24] and [25] implicitly assume some form of PU–SU coexistence to be already present, once the channel has to be utilized by the admitted SUs.

Other coexistence research: Our motivation is closer to the work of Geirhofer *et al.* [26], [27] that identifies 802.11 as a coexistence system in time domain. But it did not focus on incumbent protection that is central to DSA.

Protocol-specific coexistence has also been suggested in literature. Apart from the obvious disadvantage of applying only to specific wireless standards, such approaches also have limited application across the spectrum. For instance, coexistence of WiMAX (802.16) and WiFi (802.11a) has been considered in [28]. Such protocol-specific heterogeneous coexistence approaches typically require non-trivial modification to all the protocols.

The unlicensed coexistence approach proposed in [29] relies on explicit PU–SU communication or additional infrastructure to enable DSA coexistence. Clearly, this approach forces

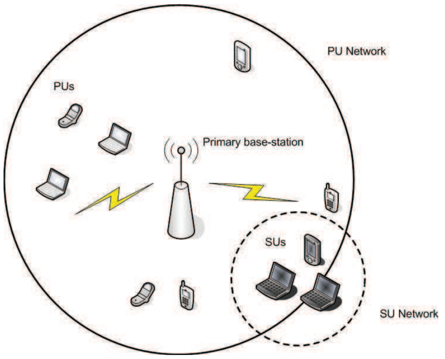


Fig. 1: Typical DSA deployment model—the circles denote the transmission range of the PU base-station and SUs.

incumbents to adapt for DSA, which may be unacceptable to licensees. In contrast, our DSA coexistence solution makes no such limiting assumptions and runs without cooperation from incumbents.

III. Preliminaries

A. System Model

We consider secondary wireless networks that consist of SU devices which are equipped with a DSA-capable wireless card (e.g., designed on a software-defined radio platform) and other necessary hardware, like a wide-band antenna (e.g., 400MHz-3GHz wide). We assume that SUs have one data transceiver for design simplicity and cost-effectiveness. Nevertheless, our approach also works for multi-transceiver systems.

A SU network dynamically tunes to a licensed channel that may "belong" to a co-located PU network. An example scenario is shown in Fig. 1. We assume that all SU devices in the SU network have a similar view on PU activity. This is reasonable because DSA protocols ensure that all SUs are coordinated, e.g., via control channel sensing exchanges [20], [21]. However, this assumption is not strictly necessary (details in Section VII).

We do not assume any type of active coordination between PUs and SUs for coexistence. This ensures that the proposed DSA coexistence can be widely deployed, even alongside legacy PU systems.

B. Role of Spectrum Sensing

The effectiveness of DSA coexistence is highly dependent on how correct and timely is its knowledge about the underlying channel conditions, especially PU activity. High-fidelity spectrum sensing requires *Quiet Periods* (QPs), during which no SU should engage in any transmission. A QP may vary from less than 1ms (using just energy-detection in high frequency bands) to 100ms or more (using feature-detection) [16]. With advances in sensing technology, QP duration is expected to decrease in future. Still, QPs induce a significant overhead in DSA if scheduled frequently. Our solution would only benefit from advances in the field of spectrum sensing.

Though an external sensing infrastructure (e.g., sensing sensor network, or offline channel geolocation databases as used in [30] and recommended in [5] for TV spectrum) simplifies the sensing task, it must still be performed in real time for multiple reasons, e.g., to dynamically check for spectrum white spaces, detecting unexpected incumbent transmissions, and coexistence. While fixed database based sensing and DSA may work reasonably well for slow-varying TV spectrum, it can be very restrictive in wireless channels used for consumer communication, as channel utilization cannot be changed dynamically, even for PUs. The underlying tradeoff between spectrum sensing and performance necessitates intelligent scheduling of transmission and sensing, which is a key design goal of our DSA coexistence.

IV. Motivation

With a view on emerging *secondary markets* [31], upcoming licensed communication technologies (e.g., protocols for 4G and beyond) are being designed to explicitly permit unlicensed coexistence of foreign communication protocols [32], such as IEEE 802.16h [8]. We argue that the onus is on DSA technology to demonstrate that it can provide performance gains to the secondary user at an acceptable interference to primary users.

Spectrum surveys [3], [9] show that wireless channels are severely underutilized (average <50% across 20MHz-3GHz spectrum), even in urban areas, along both frequency and time domains. Further, [9] indicates that available spectrum white spaces are not probabilistically independent, e.g., their durations show a high degree of temporal correlation (on average > 0.7). This is shown to hold true even in relatively higher utilized and fast-varying channels, like cellular/3G service channels.

A high degree of temporal correlation in time-domain spectrum white spaces [9], forms the basis for our DSA coexistence solution. Although individual spectrum white space durations may be random, they are correlated to recent past and near future channel occupancy state of PUs.

With the above motivating factors, we now focus on the technical aspects of DSA coexistence. In addition to the stringent requirements of DSA (PU-Safety vs. SU-efficiency), generic DSA coexistence solution must also account for the diversity in incumbent channel-access behavior. We demonstrate this challenge through the following experiment.

A. Experimental Verification

To verify the existence and impact of SU-efficiency vs. PU-safety tradeoff, we conducted simple tests using a modified MadWifi device driver with Atheros based 802.11 wireless cards. The PU follows ON/OFF channel access behavior with each ON/OFF duration being exponentially random. We changed the PU's average ON/OFF duration while keeping the the average availability of channel white spaces at 50%. We point out that ON/OFF model is a well-accepted model for incumbent behavior in DSA research [1], [18].

We used an ideal DSA MAC (modified 802.11 MAC) that can always detect PU signals by simple carrier sense when no data is being transmitted by the interface (i.e., no sensing overhead accrued). The modified MAC mirrors the

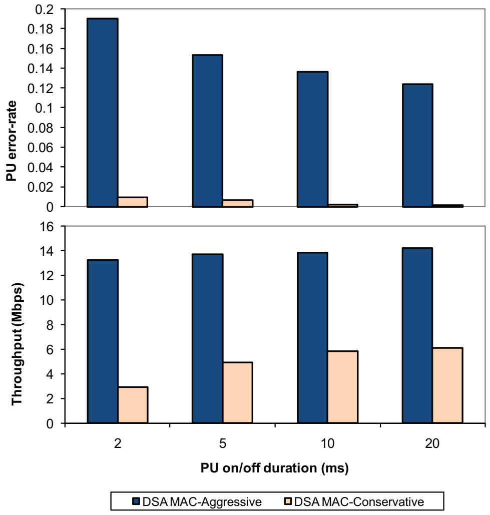


Fig. 2: Performance with current DSA coexistence model.

state-of-art DSA coexistence strategy followed in the UHF spectrum: it halts transmission when the PU signal is detected (within a time-limit), and attempts to transmit when sensing declares the medium to be free of incumbents. To show that any straightforward modification (e.g., reducing time-limit for stopping secondary transmission) of this simple coexistence strategy is insufficient in fast-varying channels, we designed two types of DSA MAC—aggressive and conservative. The two MAC types differ only in backoff duration before starting transmission.

In the aggressive MAC, SUs access the channel with a smaller backoff time. In this case, as shown in Fig. IV-A, interference to the PUs can be as much as 19% of its channel-access time. In contrast, SUs access the channel with a large backoff time when using the conservative MAC. Here, the error-rate would be much more acceptable ($< 2\%$). However, SU-efficiency then degrades by more than 75% as compared to the aggressive MAC.

In this experiment, the average amount of available channel white spaces (the OFF periods) is same (50%) for both fast-varying (e.g., PU ON/OFF=2ms/2ms) and relatively slow-varying channels (e.g., PU ON/OFF=20ms/20ms). The key difference is that it is much more fragmented in time-domain for the case of fast-varying channels. With current DSA coexistence schemes, such fast-varying channels will always be seen by SUs as occupied by the PUs and hence un-utilized, even though the channel is free 50% of the time.

An option for the SUs is to switch to a different channel, as currently followed in UHF spectrum. However, as mentioned earlier, this is a task of frequency-domain spectrum management, which is orthogonal to time-domain DSA coexistence considered here. Nevertheless, changing channels would not eliminate the aforementioned problem, if similar PU behavior is exhibited on the new channel. Note that closely-spaced channels in the licensed spectrum, in general, exhibit similar characteristics as they are used for similar services.

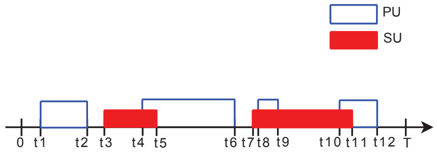


Fig. 3: The blocks represent the medium access durations by either SU (shaded) or PU (unshaded). Simultaneous access results in interference to PUs from SUs.

V. Problem Statement

The following formulations are in the context of a DSA coexistence duration, say $[0, T]$ shown in Fig. 3.

CGF description: To mathematically quantify the tradeoff between PU-safety and SU-efficiency—the two key requirements of DSA coexistence, we first model the active channel-usage intervals as a set of ordered pairs:

$$CUI = \{(t_i, t_j)\text{: channel use from } t_i \text{ to } t_j, t_i < t_j\}. \tag{1}$$

Each element of the set *CUI* represents a finite time-interval when the channel was utilized. Also, if $(t_a, t_b) \in CUI$ and $(t_c, t_d) \in CUI$, then $t_a < t_c \implies t_b < t_c$, and vice versa.

We define two parameters:

- $I_{ps}(CUI_{PU}, CUI_{SU})$: *PU–SU Interference Factor*, or the maximum fraction of PUs' transmission time interfered by SUs' transmissions, $0 \le I_{ps} \le 1$.
- $U_s(CUI_{PU}, CUI_{SU})$: *SUs' Channel Utilization Factor*, or the fraction of time utilized by SUs ($0 \le U_s \le 1$).

For instance, in Fig. 3,

$$\begin{aligned} I_{ps} &= \frac{(t_5 - t_4) + (t_9 - t_8) + (t_{11} - t_{10})}{(t_2 - t_1) + (t_6 - t_4) + (t_9 - t_8) + (t_{12} - t_{10})}, \\ U_s &= \frac{(t_5 - t_3) + (t_{11} - t_7)}{T}. \end{aligned} \tag{2}$$

The *Coexistence Goodness Factor* (CGF) is a 2-dimensional metric defined as

$$CGF(CUI_{PU}, CUI_{SU}) = (I_{ps}, 1 - U_s). \tag{3}$$

CGF is a significant improvement over the state-of-art DSA coexistence metric of absolute time-bound for limiting interference to PUs. The improvement comes for two reasons—(a) PU-safety is extended to include the cumulative impact of SU transmissions throughout coexistence period; (b) Overall SU-efficiency is also incorporated.

Note on PU–SU interference model: One of the fundamental characteristics of wireless communication is that detecting collisions while transmitting is extremely difficult (without external aid) due to very high transmit energy density around the transmitter. Thus, it is impossible to ensure perfect PU-safety in DSA in current wireless communication model. Still, minimizing interference to PUs must be a key goal for any DSA coexistence scheme to make DSA more useful and acceptable to licensees. CGF metric incorporates this objective comprehensively.

Channel White-Space Utilization Problem: The goal of a DSA coexistence is to minimize CGF. Since the two individual objectives in CGF conflict with each other, we formulate the following multi-objective optimization problem (MOP). For CUI_{PU} during $[0, T]$,

$$\min_{CUI_{SU}=\{(t_i,t_j):0\leq t_i<t_j\leq T\}} CGF(CUI_{PU}, CUI_{SU}). \qquad (4)$$

The solution space of the above MOP consists of the possible channel-access schedules for the SU network. Eq. (4) can be shown to be Pareto-optimal, with the optimal value depending on CUI_{PU}. The optimal value for I_{ps} objective is always 0. The ideal solution corresponds to perfect usage of the channel by the SUs—100% utilization of the spectrum white spaces on the channel, with zero SU transmission during PU ON periods. In theory, it is simple to solve the optimization problem in Eq. (4) using the well-known MOP optimization techniques like the Aggregate Objective Function (AOF) or Normal Boundary Intersection (NBI) method.

Need for a practical approach to DSA coexistence: In practice, it is difficult to optimize Eq. (4) for a number of reasons, arising due to lack of real-time knowledge about operating parameters during deployment. First, CUI_{PU} is most likely unknown to the SUs (PU–SU cooperation cannot be assumed). Second, SUs need to schedule their transmissions in real time which is affected by other channel-related factors, thus preventing maximal utilization of PU-free intervals or white spaces. Third, DSA is inherently inefficient as it needs to schedule sensing (and other events) that prevents full utilization of channel white spaces.

VI. SpeCWiFi

Our solution to the channel white-space utilization problem involves incumbents' channel-usage pattern estimation with joint sensing and transmission scheduling.

We present our approach in the context of SpeCWiFi—which enhances the 802.11 DCF with our DSA coexistence scheme, while maintaining its distributed operation semantics. SpeCWiFi serves as a concrete illustration of how to apply our proposed DSA coexistence protocol. Our goal is not to build a full DSA MAC protocol as in [4], [20]. Rather, our focus is on the DSA coexistence problem, which is typically a component of the DSA MAC.[3]

Further, it must be emphasized that *the proposed DSA coexistence is generic* and can be incorporated as part of *any* DSA protocol even though we develop it for WiFi-type system for implementation and evaluation tractability. Also, for SpeCWiFi, SU–SU coexistence (or self-coexistence) is handled by the existing CSMA/CA mechanism in the 802.11 MAC.

A. Adaptive Dual-mode Licensed Operation

We propose a dual-mode DSA coexistence scheme, consisting of *Safe Mode* (SM) and *Aggressive Mode* (AM), as shown in Fig. 4. Normal mode corresponds to operations in unlicensed (or *home*) channels, where DSA coexistence does not apply.

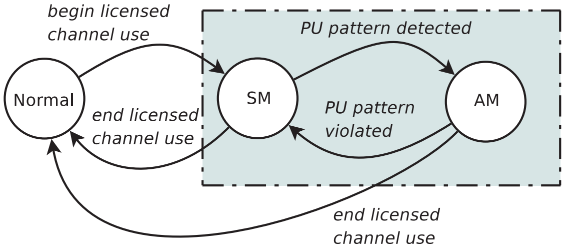


Fig. 4: State-transition diagram of DSA coexistence.

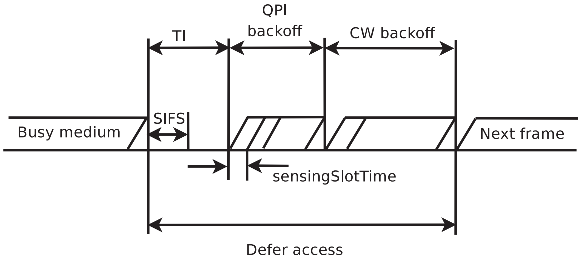


Fig. 5: Channel access in the Safe Mode of SpeCWiFi.

B. Safe Mode (SM)

SM is the default mode of SUs when operating in licensed channels. Once a SU network enters a licensed channel, it starts operations in SM, and may switch to AM when a PU channel-usage pattern is detected, as shown in Fig. 4. Conversely, the SU network will switch from AM to SM, if the expected PU channel-usage pattern is violated. The philosophy behind SM is to "transmit less, observe more." This allows SUs to continuously gather sensing information without too many time-gaps. A high-quality sensing time-series is crucial to the determination of PU channel-usage patterns (see Sections VI-D and VI-E).

We define an *Atomic Packet Exchange* (APE) as a sequence of frame exchanges resulting in a complete transfer of a set of MSDUs from the sender. In SM, time-consuming APEs such as burst-type exchanges and prioritized access are prohibited in order to prevent SUs from using the licensed channel for a long time in one stretch. Regular APEs are allowed, with the condition that the APE duration must confirm to the DSA regulatory guidelines on incumbent detection.

Every APE is followed by a *Quiet Period Interval* (QPI), before the channel can be accessed for the next APE. Similar to 802.11 contention window variation [33], QPI varies according to:

$$QPI = QPW \times sensingSlotTime \qquad (5)$$

where *QPW* is the quiet period window and takes an integer value in the range over the interval [1, QPW_{max}]. The minimum duration adequate for high-fidelity spectrum sensing is

[3] Issues such as when SpeCWiFi should switch between licensed and unlicensed operations or which channel to select, are unrelated to the PU–SU coexistence problem addressed here. Interested readers are referred to prior work on DSA spectrum management in Section II.

indicated by *sensingSlotTime*, and its value is a fixed input derived from the sensing technology used.

QPW (or equivalently QPI) is varied based on recent sensing observations in order to adaptively balance the SU's need for sensing opportunities vs. data transmission. The initial value of QPW is QPW_{max}. For every QPI result indicating the PU's absence, QPW is reduced by half. Once QPW reaches 1, it remains at this value until it is reset. Thus, even in SM, data transmission can be frequent when PUs are not observed on the channel for a long time. If a PU is detected during the QPI, QPW is reset to QPW_{max}. QPI must then be re-initialized. Thus, a recent PU detection makes SUs wait longer before attempting to transmit even when the medium may be sensed to be currently free, as the PUs could likely be engaged in an ongoing communication session.

In wireless networks, after every packet transmission, a sufficient turnaround time is required for decoding and resetting interfaces (e.g., SIFS in 802.11). In SpeCWiFi, QPI follows the *Turnaround Interval* ($TI > SIFS$) after each APE, and proceeds with the CW backoff period as shown in Fig. 5. Further, although QPI is calculated individually by the SU nodes (using Eq. (5)), they converge to the same value and are scheduled at the same time. This is because all the nodes of a SUG have a consistent view about channel conditions in terms of PU detection (see Section III-A), and also follow the same channel access model. Typically, the coordination component in most DSA MAC/PHY protocols ensures that nodes have a similar view of the channel (e.g., via control channel exchanges as in [20], [21], or beacons as in [6]). By leveraging this feature, the proposed SM operation is applicable to multiple collision domains.

We briefly address the rare case when explicit spectrum coordination is absent from the parent DSA MAC protocol in Section VII.

C. Aggressive Mode (AM)

In case there is a pattern of PUs' ON and OFF durations, SUs may not need to waste QPIs that could otherwise be utilized for data transmission. To achieve better CGF, AM attempts to exploit occurrence of PU channel-access pattern, which can exist due to: (a) explicit design in the incumbent wireless protocol to facilitate unlicensed coexistence (as in 802.16h [8] or upcoming 4G systems), (b) traffic characteristics in the incumbent network (as in TDMA-based licensed networks where some slots go unused [19]), (c) channel reservation semantics of the licensed channel's owner/operator [34].

In contrast to SM, the principle of AM is to "transmit more, observe minimally." In AM, the channel-usage pattern of PUs is known based on sensing observations gathered in SM. Details on how to estimate the PUs' usage pattern will be provided in Section VI-D.

In AM, QPIs are scheduled at frequency f_{qpi} in order to ensure the periodic sensing required to ascertain any out-of-pattern PU traffic. The frequency f_{qpi} must conform to the regulatory time guidelines in terms of detecting any PU transmission. Any unexpected detection of PU traffic will result in the PU channel-usage pattern violation, and the SUs switch back to SM. Since QPI is scheduled relatively infrequently in AM, the QPW value is fixed at QPW_{max} allowing maximum duration for each QPI to enable more reliable sensing.

D. Estimation of PU Channel-Usage Pattern

The sensing component of DSA provides information on whether incumbent activity has been detected at various time instants on the licensed channel. Using bits 1 (to indicate the PU presence) and 0 (to indicate the PU absence), the sensing observations can be represented as a binary time-series:

$$s = [s_1, s_2, \ldots, s_i, s_{i+1}, \ldots], s_i \in \{0, 1\}.$$

The series s has a bounded number of elements (N) over a finite time-window. An element s_i of the series corresponds to the sensing observation taken at time instant t_i. Series s constitutes the input available for PU channel-usage pattern detection.

Many well-known techniques (e.g., genetic algorithms) have been used for pattern recognition and trend analysis, especially in data-mining and machine learning. However, they are quite complex to implement at device level, and involve significant resource overhead. Further, such approaches typically require a high degree of training and thus cannot be deployed online, such as 2-D frequent pattern mining proposed in [9]. We ruled out other high-overhead techniques here (like FFT and autocorrelation), as MAC processing needs to be agile, real time, and must operate on limited memory and computational power.

Instead, we make use of *Approximate Entropy (ApEn)* [10], [35] for pattern recognition. ApEn is a measure of regularity (or irregularity) present in a discrete sequence, e.g., binary sequences like s. Given a small number of observations, ApEn can be used to classify complex systems including deterministic and stochastic processes, without any additional information about system behavior. Hence, ApEn measure is well suited for analyzing PU channel-usage behavior. ApEn has been shown to be useful in diverse contexts, e.g., cardiovascular data analysis [36] and spectrum sensing [19]. A brief description of ApEn is provided next.

1) Approximate Entropy (ApEn): Consider the binary series s consisting of N elements or bits. ApEn is defined for each length L of consecutive bit vectors that can be constructed from s. For each vector i of length L and resolution r, its correlation sum is given by:

$$C_i^L(r) = \frac{\text{Num. vectors of length } L \text{ similar to } i}{N-L+1}.$$

Two vectors are similar when their hamming distance $\leq r$. The mean size L logarithmic correlation sum is:

$$\Phi^L(r) = \frac{1}{N-L+1} \sum_{i=1}^{N-L+1} \log C_i^L(r).$$

Finally, approximate entropy of s is defined as:

$$ApEn(L, r, N)(s) = \begin{cases} \Phi^L(r) - \Phi^{L+1}(r) & \text{, if } L \geq 1 \\ -\Phi^1(r) & \text{, if } L = 0. \end{cases} \quad (6)$$

Algorithm 1 ApEn calculations for s

```
Require: $s = [s_1, s_2, \ldots, s_N]$, $L_{max}$
Require: Maximum expected pattern length $L_{max}$
Ensure: $L_{max} + 1 \leq N$
Declare ApEn array $ApEn[L_{max}]$ {0-indexed}
Declare logarithmic correlation array $\Phi[L_{max}]$
$\Phi[0] \leftarrow 0$ {Initialize boundary condition}
$L \leftarrow 1$ {Initialize pattern length to 1}
while $L \leq L_{max} + 1$ do
    $\Phi[L] \leftarrow 0$
    Declare distance array $d[N - L + 1][N - L + 1]$
    for $i \leftarrow 1$ to $N - L + 1$ do
        for $j \leftarrow 1$ to $i$ do
            if $i = j$ then
                $d[i][j] \leftarrow 0$
            else
                $d[i][j] \leftarrow \max_{k \leftarrow 1,2,\ldots,L}[|s_{i+k-1} - s_{j+k-1)}|]$
                $d[j][i] \leftarrow d[i][j]$ {Using symmetry of $d[i][j]$}
            end if
        end for
    end for
    Declare correlation vector $C[N - L + 1]$
    for $i \leftarrow 1$ to $N - L + 1$ do
        $C[i] \leftarrow 0$
        for $j \leftarrow 1$ to $N - L + 1$ do
            if $i \neq j$ && $d[i][j] \leq 0$ then
                $C[i] \leftarrow C[i] + \frac{1}{N-L+1}$ {Calculate $C_i^L(0)$}
            end if
        end for
        $\Phi[L] \leftarrow \Phi[L] + \frac{C[i]}{N-L+1}$ {Calculate $\Phi^L(0)$}
    end for
    $ApEn[L-1] \leftarrow \Phi[L-1] - \Phi[L]$ {Calculate $ApEn(L-1, 0, N)$}
    $L \leftarrow L + 1$
end while
```

ApEn indicates the degree of regularity present in the sensing information s. As can be seen from Eq. (6), $ApEn \leq 1$. Large values of ApEn (e.g., 0.9) denote high irregularity in s, while small values of ApEn (e.g., 0.1) point to the presence of a regular pattern in s. In this context, ApEn predicts the probability of any pattern in s. We use $r = 0$ in order to ensure the strictest comparison of vectors in s for more reliable pattern detection.

2) Algorithms: Our pattern recognition method is based on parametrized decision-making. We present Algorithms 1 and 2 that jointly accomplish this task.

Algorithm 1 encodes efficient calculation of ApEn values for the sensing time-series s. The output of Algorithm 1 is $ApEn[L_{max}]$, an array of ApEn values. Algorithm 2 takes this array together with $ApEn_{thresh}$ as input parameters to decide, based on comparisons with $ApEn_{thresh}$, whether a pattern is present in s. If $ApEn(L, 0, N) \leq ApEn_{thresh}$, a pattern of length L is present in series s. The length of the best recognized pattern is the output of Algorithm 2.

Algorithm 2 Pattern recognition decision-making

```
Require: $ApEn[L_{max}]$, $ApEn_{thresh}$
$ApEn_{min} \leftarrow \infty$ {Initialization and boundary cases}
$L_{pattern} \leftarrow -1$
$Found \leftarrow FALSE$
for $i \leftarrow 1$ to $L_{max}$ do
    if $ApEn[i] \leq ApEn_{thresh}$&&$ApEn[i] \leq ApEn_{min}$ then
        $ApEn_{min} \leftarrow ApEn[i]$
        $L_{pattern} \leftarrow i$
        $Found \leftarrow TRUE$
    end if
end for
return $Found, L_{pattern}$
```

3) Correctness and Complexity: Algorithm 1 is easily proved correct as it is based on Eq. (6). The asymptotic worst-case runtime complexity of Algorithm 1 is $O(L_{max}^2 N^2 + L_{max}^3)$. When the length of the detected pattern is small, L_{max} is effectively a small constant. Hence, the asymptotic runtime is simply $O(N^2)$. Alternatively, if the length of the pattern is large (or $L_{max} \sim N$), then the runtime is $O(N^4)$. The asymptotic space complexity of Algorithm 1 is $O(N^2 + L_{max}^2)$, or $O(N^2)$ in the worst case.

The correctness of Algorithm 2 is easily proved by using contradiction. Algorithm 2 is a linear-time algorithm with the worst-case complexity as $O(L_{max})$. For small values of L_{max}, it is essentially a constant-time algorithm. Also, if $L_{max} \sim N$, then the runtime complexity is $O(N)$. The asymptotic space complexity of Algorithm 2 is the same as its runtime complexity.

4) Applying the algorithms to AM: Given the PU-channel usage pattern of length $L_{pattern}$ (from Algorithm 2), the SUs estimate the start-time and duration of each PU-free and PU-busy periods (relative to the current time) as follows.

Let t_i be the time instant when sensing observation s_i was made. The key insight is that the most recent sensing observations $S_{pattern} = \{s_1, s_2, \ldots, s_{L_{pattern}}\}$ will repeat over the next $T_{pattern} = t_{L_{pattern}} - t_1$ time-interval. Thus, the PU-free/busy periods are estimated based on the elements of $S_{pattern}$ and the difference between their observation time. For instance, if the values of successive pattern elements $\{s_i, s_{i+1}\}$ are $\{0, 0\}$, then PU-free duration of $t_i - t_{i+1}$ is predicted. Similar is the case for $\{1, 1\}$ where PU-busy period is predicted. For observations of type $\{0, 1\}$ or $\{1, 0\}$, the transition is assumed to be midway between the individual observation times. Here, the exact transition times depend on the sensing frequency and sensing duration, and may not be known. Thus, any conflict during the initial phase of such transitions are ignored.

As an estimated pattern may not be 100% accurate, there must be a reasonable margin for error-tolerance. Any mismatch in prediction and expectation of PU-free/busy period results in a probabilistic switch to SM based on $ApEn_{thresh}$ value. Thus, if the fraction of mismatches observed surpasses

$ApEn_{thresh}$, the SUs switch back to SM.

E. Sensing: Quality vs. Quantity Tradeoff

As noted earlier in Section III-B, sensing information is crucial for DSA coexistence, especially for the PU channel-usage pattern estimation. Apart from a reliable sensing technology, the overall sensing data should be of sufficiently high granularity to detect patterns over short durations. Thus, frequent scheduling of sensing quiet periods is key to high quality sensing information.

In SM, the sensing is quite frequent—after every APE, and hence, PU activity can be captured at fine granularity. However, frequent sensing quiet periods can incur a high overhead for communication. Thus, sensing is scheduled on a *as-needed* basis when the PU channel-usage pattern is known.

F. Remarks

Practical considerations: In practice, additional minor changes to DSA MAC protocols may be required in order to incorporate our approach. For instance, to facilitate mode management (e.g., during node association to an AP), the DSA MAC should include mode information in control packets. Our SpeCWiFi implementation includes mode information in the MAC header.

Why pattern detection is useful and applicable?: As discussed earlier, traditional reactive coexistence approach does not scale in channels exhibiting spectrum white spaces of very small duration (order of few milliseconds). Studies [9] have shown that such licensed channels (20MHz–3GHz) are characterized by highly correlated time-domain white spaces in numerous spectrum regions (on average > 0.7). Majority of such patterns are seen to last for several minutes or more. Besides the implicit usage pattern present in current licensed spectrum, certain upcoming licensed protocols (like IEEE 802.16h) feature explicit regular white spaces.

On interference to PUs: It is extremely difficult to guarantee zero interference to PUs during deployment, as discussed in Section V. The PU channel-usage pattern may change, causing some interference while the mode is adjusted to SM. Extremely small time-scale PU transmissions may also be missed. Further, even in SM, there may be some interference to incumbents, especially if the incumbent channel-access behavior is highly random.

Therefore, our proposed DSA coexistence scheme tries to achieve the next-best goal, i.e., minimizing such interferences. In the process, our solution can provide the operational benchmark levels for wireless operators and consumers. In our evaluation (see Section IX), average interference to incumbents is found to be less than 2% of coexistence duration when PU channel utilization is 50%. Similarly, this approach may not provide 100% SU-efficiency in practice, but it is found to be sufficiently high in all cases ($> 90\%$ on average).

VII. Boundary Environment

SUs in the network may contradict to each other in sensing the incumbent's presence. This could occur, for example, in a boundary environment condition where the incumbent signal strength is very weak around the SU network such that certain SUs can sense the PU signal, while others cannot. Also, channel sensing errors can lead to such discrepancies.

Typically, this is handled through explicit coordination functions (e.g., using control channels) in a DSA MAC protocol. In the absence of an explicit coordination, we propose the following conflict resolution policy.

Considering the importance of PU-safety, conflicts between AM and SM are resolved in favor of SM. Packets should carry mode information in order to quickly identify and resolve conflicts when they occur.

As a further enhancement, on-demand control packets can be used by SUs (including passive SUs) for quick indication of the PU's presence. A SU node broadcasts a *PU On* (PUO) packet when the PU is sensed on the channel, followed by a *PU Ceased* (PUC) packet when the PU ends the transmission. To avoid overheads, PUO-PUC packet exchange must only be undertaken when a SU node detects (or expects) any SU transmission while it has knowledge of simultaneously ongoing PU activity on the channel—indicating a conflict among nodes in the SUG. Further, a randomized delay should be used to prevent multiple SUs from broadcasting PUOs at the same time.

VIII. Implementation

We have implemented SpeCWiFi by augmenting the open-source MadWifi driver (madwifi-0.9.4) [11] to develop a software prototype that operates Atheros 802.11 wireless cards. Ideally, many of the SpeCWiFi features should have a hardware systems-on-chip implementation for precise and real-time MAC operations, similar to the current network interface cards available in the market. However, the absence of any suitably-priced hardware with required performance led us to implement SpeCWiFi (including sensing emulation) within the kernel device driver. Thus, SpeCWiFi operations may not be as quick as they can be, since the driver has to share the processor with other processes. While the implementation platform is not the best for SpeCWiFi, it is found to be more than adequate for analyzing performance trends in realistic deployments.

Again, we emphasize that even though our prototype implementation is based on a WiFi-type platform, the proposed DSA coexistence is applicable to any DSA protocol and across most licensed spectrum.

A. SpeCWiFi SU

The SpeCWiFi implementation consists of its state machine together with its access model (see Figs. 4 and 5). To compensate for hardware behavior that confirms to the 802.11 standard, certain approximations were utilized as detailed below.

To emulate the sensing QPI, hardware queues' congestion window parameters were updated, depending on the QPI delay needed (*cwmin* = *cwmax* = k). Although this does not ensure exact delay every time (random *cwnd* could not be disabled as it is performed in hardware), it ensures that there is an average delay proportional to k. The QoS control bits in the standard 802.11 MAC header are used to carry the DSA coexistence

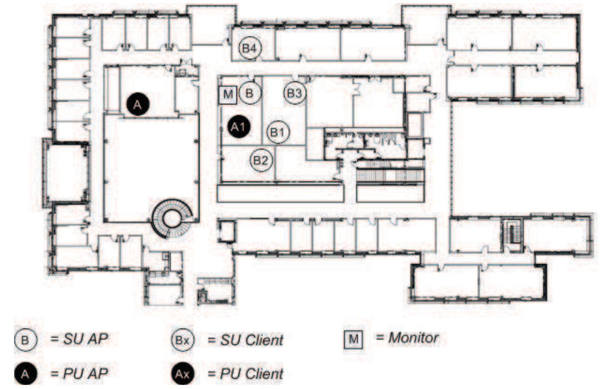


Fig. 6: SpeCWiFi testbed in the building floor.

mode information. RTS/CTS packets emulate PUO/PUC packets that are used in a boundary environment (see Section VII). We did not use the control-channel approach, as it would have required extensive changes to the MadWifi driver and Atheros firmware. High resolution timers *hrtimers* were utilized to provide sufficient resolution with high precision interrupts.

B. PU Emulator

Lack of transmission license in licensed channels and unavailability of inexpensive testing equipment forced us to emulate PU signals. We use click-1.6.9 modular router [37] together with MadWifi. Click is configured to generate the desired burst of packet streams to emulate a PU's ON period, while there is no transmission during OFF periods. The key challenge in emulating PUs using 802.11-based cards, is to prevent carrier sense during the PU's ON time. This is accomplished by setting appropriate MadWifi configuration parameters (e.g., *TXOP backoff* is disabled), and by ensuring the transmission range asymmetry between PUs and SUs.

IX. Evaluation

A. Testbed Setup

Our testbed was set up according to the system model (see Figs. 1 and 6). The setup consists of one SU network and one PU network. As mentioned earlier, self-coexistence (SU-SU coexistence) is easily achieved through existing mechanisms, e.g., DCF in 802.11 MAC. For microbenchmarks, only a single AP–client pair is active in the SU network, while multiple pairs are used for macrobenchmark experiments. The machines are Dell Inspiron 600m laptops with Linksys A+G wireless cards, and running Ubuntu 8.04 Linux (2.6.24-23 kernel). We used 802.11a channel 36 (5.18GHz) for our experiments as it was found to be free of other interfering devices in our experimental setting.

Netperf with saturated UDP stream is used to fully stress out the prototype implementation in microbenchmark experiments, while traces are utilized to study coexistence performance in a more realistic setting. The default sensing granularity (or *sensingSlotTime*) used is 1ms and the default PU pattern consists of exponentially random ON/OFF = 5ms/5ms (on average) durations.[4] Although there is a regular pattern in PU channel-usage (reflecting the temporally correlated white spaces in licensed spectrum), the ON/OFF periods are themselves random. Other default parameter values are: $N = 100$, $L_{max} = 50$, $ApEn_{thresh} = 0.1$, $f_{qpi} = 0.5s^{-1}$, and $QPW_{max} = 10$.

B. Performance Metrics

CGF (in terms of I_{ps} and U_s[5]) is the main metric of evaluation of the proposed coexistence schemes via SpeCWiFi. We also show the overall end-to-end performance in terms of throughput achieved, where necessary.

C. Results and Discussion

In every CGF comparison plot, $U_{s_{max}}$ represents the maximum SU utilization and $I_{ps_{min}}$ represents the minimum interference to PUs over the coexistence duration in that setup. Achievement of $U_{s_{max}}$ and $I_{ps_{min}}$ implies perfect DSA coexistence. Note that $I_{ps_{min}}$ is always 0.

First, we discuss the important microbenchmark experiments and their results.

1) Overhead characterization: To capture the run-time overhead, the testbed was run with PU turned off. This scenario also establishes the performance benchmark when PUs are not present on a licensed channel for long durations (e.g., in slow-varying channels). Fig. 7a shows the average throughput obtained with a 95% confidence interval over every 10s for the first 50s. The initial throughput is slightly lower than the later values for SpeCWiFi, as it starts out in SM and takes few hundred ms to switch to AM. The throughput quickly stabilizes to around 29.5Mbps as seen from the graph. Overall, SpeCWiFi throughput is 0.6Mbps less than native 802.11 throughput in an empty channel. This loss in performance ($\approx$ 2%) is attributed to the computational overhead, e.g., timer routines and ApEn calculations. As pointed out earlier, the overhead should be lower in a hardware-based implementation where the access model (including sensing computations) does not have to contend for the processor time.

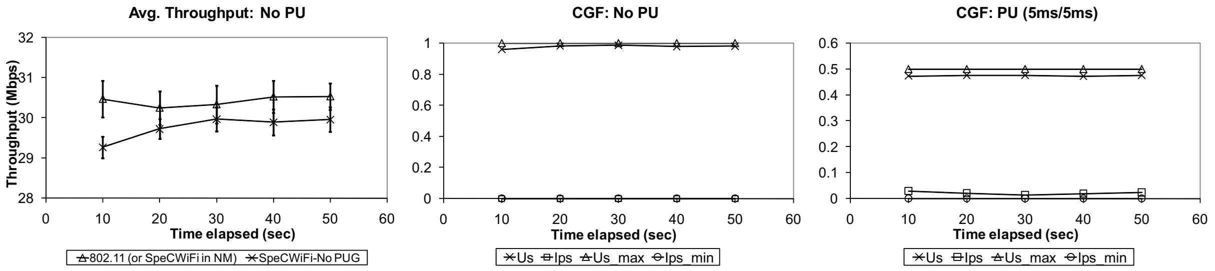
Fig. 7b shows the same result as in Fig. 7a, but in terms of average CGF. U_s is found to be very close ($\approx 98\%$) to $U_{s_{max}}$, while I_{ps} is trivially 0.

Fig. 7c shows the average CGF seen over every 10s interval, when PU is present on the channel with 50% channel-usage (average PU ON/OFF=5ms/5ms). SpeCWiFi manages to achieve an average of 96% utilization ($U_s \approx 0.48$ or 14.4Mbps throughput) of the available channel white-spaces, with less than 2% rate of interference to PUs.

2) Impact of parameters: Fig. 8a shows the impact of history size $N (= 2L_{max})$ on CGF. The sensing granularity is 1ms. For very low values for history ($N < 20$), CGF is found to be significantly lower—with high I_{ps} and low U_s. The main reason for this is an insufficient history window to capture the full PU pattern (5ms/5ms ON/OFF), resulting in frequent mode

[4] Results for uniformly random and log-normally distributed ON/OFF durations are found to be similar, and omitted due to lack of space.

[5] We plot U_s instead of $1 - U_s$ to simplify plot visualization and to show its direct correlation to the throughput.

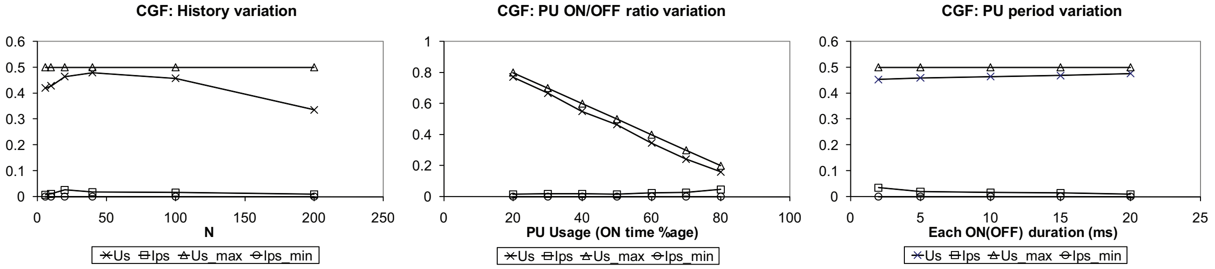


(a) Throughput (No PU)

(b) CGF (No PU)

(c) CGF (with PU)

Fig. 7: Performance variation over time.



(a) History Window

(b) PU channel-usage

(c) PU pattern

Fig. 8: Impact of parameters and incumbent behavior I.

switches. Similarly, the performance is poor when the history size is large (beyond ≈150). However, in this case we observe that the computational overhead is the bottleneck, highlighting the importance of low computational complexity of pattern estimation algorithms. The optimal value for history size in terms of L_{max}, as seen from Fig. 8a, is around the size of the PU pattern. The optimal history size must be greater than the total PU pattern window (≈20ms in this case). Thus, prior knowledge about the range of expected incumbent patterns in a spectrum region (e.g., through an external database as in [38]) may be useful in dynamically updating L_{max}.

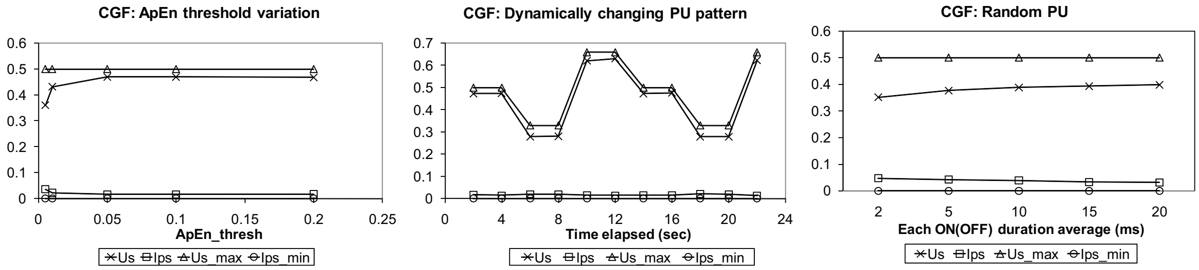
Fig. 8b shows the impact of varying PU channel-usage pattern in terms of its ON time percentage. One ON+OFF pattern lasts for 20ms. SpeCWiFi is shown to be consistent in achieving high U_s and low I_{ps}. I_{ps} is seen to be somewhat higher when the PU ON time fraction is very high. This observation suggests that in such cases, it may be better to switch to a freer channel. However, such a decision is up to the spectrum management of DSA, which is outside of the scope of this paper.

Fig. 8c shows the impact of varying PU period in terms of its ON/OFF duration, while keeping the overall PU channel-usage fixed at 50%. The graph shows that CGF is better when each ON/OFF duration is larger. However, SpeCWiFi is able to keep CGF low ($U_s = 0.44$, $I_{ps} = 0.04$) even in the presence of very fast-varying PU with average ON/OFF period of 2ms. Clearly, SpeCWiFi improves coexistence performance significantly in comparison with the traditional DSA (see Fig. IV-A)—85% improvement in PU-safety and 150% gain in SU-efficiency.

Fig. 9a shows the impact of $ApEn_{thresh}$ parameter. It shows that when a reliable PU pattern (5ms/5ms in this case) exists, there is little effect of varying the ApEn threshold. According to our observations, when there is high regularity in the sensing time-series, ApEn values tend to be very low (< 0.01), while rising rapidly on irregularity, which reflects its logarithmic nature. This fact is visible in the graph, where performance degrades only when $ApEn_{thresh}$ values are very small. In such a setting, slight fluctuations in sensing time-series may lead to mode switches (and associated overheads). Hence, keeping $ApEn_{thresh}$ value very low is not recommended.

3) Dynamic and random PU pattern: Fig. 9b shows the adaptability of SpeCWiFi when the PU channel-usage pattern changes dynamically. In this scenario, the average PU ON/OFF duration changes from 5ms/5ms to 10ms/5ms to 5ms/10ms every 4 seconds in a cyclic fashion.

Fig. 9c shows SpeCWiFi performance when the PU channel-usage pattern itself is uniformly random, i.e., the exponential distribution parameters for average ON/OFF interval are not fixed but uniformly random between 0-10ms (average = 5ms) for each ON or OFF duration. As seen from the graph, SU-efficiency is lower as compared to periodic PU scenarios. SpeCWiFi is unable to find a consistent pattern from its sensing observations and hence, remains mostly in SM. Also, PU-SU collisions increase when incumbent exhibits random behavior. However, SpeCWiFi still achieves good PU-safety ($I_{ps} < 4\%$), with reasonable SU-efficiency (average $U_s \approx 0.40$ or 80%). Similar results are obtained when the incumbent accesses the channel using other distributions to randomize the access pattern.



(a) $ApEn_{thresh}$ (b) Dynamic PU (c) Random PU

Fig. 9: Impact of parameters and incumbent behavior II.

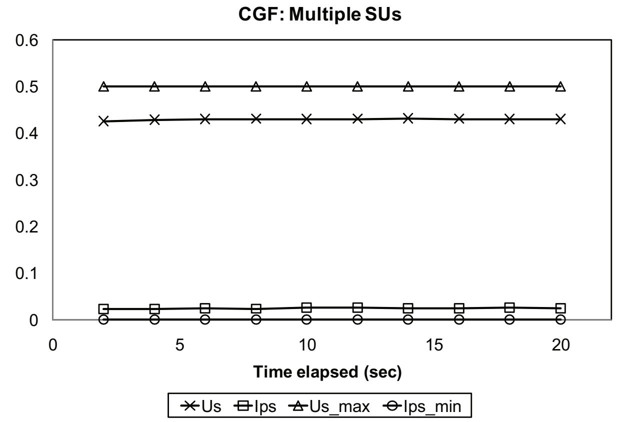


Fig. 10: Performance with multiple nodes in SU network.

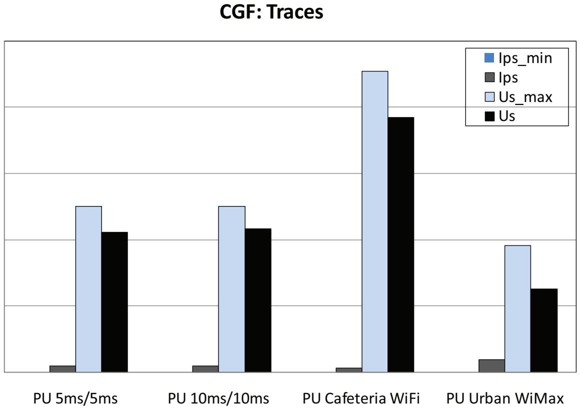


Fig. 11: Performance with traces. $I_{ps_{min}} = 0$ in all cases.

4) Macrobenchmarks and real network traces: Fig. 10 shows SpeCWiFi performance when there are 5 contending SU transmitters (4 clients+1 AP) constituting the SU network, with average PU ON/OFF pattern as 5ms/5ms. This experiment is intended to test distributed operation of the proposed DSA coexistence scheme. Self-coexistence among SUs is managed through CSMA/CA. During the experiment 2 clients were mobile, and sometimes moved beyond the range of PU transmission for short durations ($\sim$6s) to generate the boundary environment situation. As seen from the graph, PU-safety is similar to the single transmitter scenario. SU-efficiency is lower, mainly because of handling the frequent boundary situations.

For a similar topology setup as above, Fig. 11 shows the average CGF using publicly available traces to generate real network traffic load and licensed channel characteristics. Here, SUs generate packets for 30s based on snapshots of the traces for OSDI 2006 WiFi traces [39]. The packets are backlogged at the transmitter to produce a saturated stream at the channel-level in order to stress our coexistence scheme. SpeCWiFi manages to keep PU interference low (<1.8% of channel-time) with high rate of utilization ($\approx$85%), in the presence of periodic incumbents (PU ON/OFF = 5ms/5ms or 10ms/10ms), as seen in Fig. 11.

To emulate consumer licensed network, we generate two scenarios of PU channel activity—(a) based on WiFi traces from the hotspots in a cafeteria environment [40], and (b) using WiMAX traces from urban deployment (campus, subway, bus) [41]. Sufficient temporal patterns exist in such channels that allows for viable DSA coexistence using SpeCWiFi. Even for urban WiMAX case, I_{ps} is lower than 4%, with $U_s \approx 0.26$ or 65% of available opportunities. This result indicates the ability of SpeCWiFi to perform well in realistic deployments.

5) Remarks and Conclusion: We had to omit numerous results due to lack of space, but we would like to briefly mention some important observations. Our timing analysis shows that with default values of parameters (see Section IX-A), about 85% of the time is spent in AM. Also, with 50% channel usage by PUs, TCP stream utilization (U_s) is around 0.32 ($\approx$ 9Mbps), with less than 2% interference to PUs. Overall, SpeCWiFi is shown to achieve efficient and safe DSA coexistence despite diversity in incumbent characteristics.

X. Conclusion

This paper explored the feasibility of achieving general, safe, and efficient time-domain coexistence of SUs on an unlicensed basis along with PUs in a licensed channel. Key shortcomings in current PU–SU coexistence mechanisms, especially while operating in licensed spectrum characterized by fast-varying PU access, were identified. A simple, yet generic formalization of the coexistence problem was presented along with a new concept of *Coexistence Goodness Factor* (CGF) as the key coexistence performance metric. The proposed coexistence solution consists of low-overhead algorithms for PU-pattern recognition based on Approximate Entropy (ApEn), and the corresponding transmission scheduling. A dual-mode MAC operation strategy was introduced to enable its practical incorporation in real systems. An implementation based on 802.11 MAC—called *Spectrum-Conscious WiFi* (SpeCWiFi), was built and evaluated. Our experimental evaluation has shown that SpeCWiFi performs well (SU utilization 96+% with interference to PUs less than 2%, for 50% PU channel-usage), indicating the feasibility of expanding DSA-based coexistence to the relatively untouched licensed spectrum bands. The results also estbalish a basis for developing revenue and business models for unlicensed operation.

Future work includes testing different PU usage-pattern estimation methods. Further experiments involving different topologies are also planned.

ACKNOWLEDGMENT

This research was supported in part by the US National Science Foundation under Grant CNS–1160775 and also by Korea MKE (Ministry of Knowledge Economy) under the ITRC (Information Technology Research Center) support program supervised by the NIPA (National IT Industry Promotion Agency (NIPA-2012-(H0301-12-2003)).

REFERENCES

[1] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty. NeXt generation/dynamic spectrum access/cognitive radio wireless networks: A survey. *Computer Networks Journal (Elsevier)*, 50:2127–2159, September 2006.

[2] S. Haykin. Cognitive Radio: Brain-empowered wireless communications. *IEEE JSAC*, 23(2):201–220, February 2005.

[3] Mark A. McHenry and Karl Steadman. *Spectrum Occupancy Measurements*. Shared Spectrum Company, 2004. http://www.sharedspectrum.com/measurements.

[4] P. Bahl, R. Chandra, T. Moscibroda, R. Murty, and M. Welsh. White Space Networking with Wi-Fi like Connectivity. In *ACM SIGCOMM*, August 2009.

[5] FCC. *Unlicensed Operation in the TV Broadcast Bands: Second Memorandum Opinion and Order*, September 2010. ET Docket No. FCC 10-174.

[6] *IEEE 802 LAN/MAN Standards Committee 802.22 WG on WRANs*. http://www.ieee802.org/22/.

[7] FCC. *FCC Adopts Rules for Unlicensed Use of Television White Spaces*, November 2008. FCC News 202/418-0500.

[8] *IEEE P802.16h/D8*, November 2008. Part16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems.

[9] D. Chen, S. Yin, Q. Zhang, M. Liu, and S. Li. Mining Spectrum Usage Data: a Large-scale Spectrum Measurement Study. In *ACM MOBICOM*, pages 13–24, September 2009.

[10] S. M. Pincus. Approximate Entropy as a Measure of System Complexity. *Proceedings of the National Academy of Sciences (PNAS)*, 88:2297–2301, March 1991.

[11] *The MadWifi Project*. http://madwifi-project.org.

[12] J. Mitola and G. Q. Maguire. Cognitive radio: Making software radios more personal. In *IEEE Personal Communications*, pages 13–18, August 1999.

[13] Jean Kumagai. Radio revolutionaries. *IEEE Spectrum*, 4(1):28–32, January 2007.

[14] K. Tan et al. SORA: High Performance Software Radio Using General Purpose Multi-Core processors. In *NSDI*, April 2009.

[15] G. Nychis, T. Hottelier, Z. Yang, S. Seshand, and P. Steenkiste. Enabling MAC Protocol Implementations on Software-Defined Radios. In *NSDI*, April 2009.

[16] H. Kim and K. G. Shin. In-band Spectrum Sensing in Cognitive Radio Networks: Energy Detection or Feature Detection? In *14th ACM MOBICOM*, pages 14–25, September 2008.

[17] R. Chandra, R. Mahajan, T. Moscibroda, R. Raghavendra, and Paramvir Bahl. A Case for Adapting Channel Width in Wireless Network. In *ACM SIGCOMM*, August 2008.

[18] H. Kim and K. G. Shin. Efficient Discovery of Spectrum Opportunities with MAC-Layer Sensing in Cognitive Radio Networks. *IEEE TMC*, 7(5), May 2008.

[19] M. Wellens, J. Riihijarvi, and P. Mahonen. Evaluation of Spectrum Occupancy using Approximate and Multiscale Entropy Metrics. In *IEEE Workshop on Networking Technologies for Software Defined Radio Networks (SDR)*, June 2008.

[20] B. Hamdaoui and K. G. Shin. OS-MAC: An Efficient MAC Protocol for Spectrum-Agile Wireless Networks. *IEEE TMC*, 7(7), July 2008.

[21] C. Cordeiro and K. Challapali. C-MAC: A Cognitive MAC Protocol for Multi-Channel Wireless Network. In *IEEE DySPAN*, pages 147–157, April 2007.

[22] A. Kumar and K. G. Shin. Application-Aware Dynamic Spectrum Access. *ACM/Springer Wireless Networks*, 18(3):257–275, 2012.

[23] A. Kumar and K. G. Shin. Managing TCP Connections in Dynamic Spectrum Access Based Wireless LANs. In *7th IEEE SECON*, pages 1–9, June 2010.

[24] L. Cao and H. Zheng. SPARTA: Stable and Efficient Spectrum Access in Next Generation Dynamic Spectrum Access Networks. In *IEEE INFOCOM*, April 2008.

[25] L. Yang, L. Cao, H. Zheng, and E. Belding. Traffic Aware Dynamic Spectrum Access. In *WICON*, November 2008.

[26] S. Geirhofer, L. Tong, and B. M. Sadler. Dynamic Spectrum Access in the Time Domain: Modeling and Exploiting White Space. *IEEE Communications Magazine*, 45(5), May 2007.

[27] S. Geirhofer, L. Tong, and B. M. Sadler. Cognitive Medium Access: A Protocol for Enhancing Coexistence in WLAN Bands. In *IEEE GLOBECOM*, pages 14–25, November 2007.

[28] L. Berlemann, C. Hoymann, G. Hiertz, and B. Walke. Unlicensed Operation of IEEE 802.16: Coexistence with 802.11(A) in Shared Frequency Bands. In *IEEE PIMRC*, pages 1–5, September 2006.

[29] *IEEE P802.11y/D11.0*, June 2008. 3650-3700MHz Operation in USA.

[30] R. Gummadi, H. Balakrishnan, and S. Seshan. Metronome: Coordinating Spectrum Sharing in Heterogeneous Wireless Networks. In *1st COMSNETS*, pages 1–10, January 2009.

[31] Spectrum Bridge. *SpecEx: The Online Marketplace for Spectrum*. http://www.specex.com/.

[32] S. Frattasi, H. Fathi, F. H. P. Fitzek, and R. Prasad. Defining 4G Technology from the User's Perspective. *IEEE Network*, 20(1):35–41, 2006.

[33] *IEEE Std 802.11 - 2007*, March 2007. http://standards.ieee.org/getieee802/download/802.11-2007.pdf.

[34] K. E. Nolan, J. Mullany, E. Ambrose, and L. E. Doyle. Value Creation and Migration in Adaptive Cognitive and Radio Systems. In *Cognitive Radio, Software Defined Radio, and Adaptive Wireless Systems*, chapter 5, pages 145–159. Springer, 2007.

[35] S. M. Pincus and B. H. Singer. Randomness and Degree of Irregularity. *Proceedings of the National Academy of Sciences (PNAS)*, 93:2083–2088, March 1996.

[36] D. Cysarz, H. Bettermann, and P. Van Leeuwen. Entropies of Short Binary Sequences in Heart Period Dynamics. *AJP - Heart and Circulatory Physiology*, 278(6):H2163–H2172, June 2000.

[37] *Click Modular Router*. http://read.cs.ucla.edu/click/.

[38] Spectrum Bridge Inc. *First White Spaces Network*, October 2009. http://spectrumbridge.com/web/images/pdfs/PR/claudville-whitespaceproject-pressrelease.pdf.

[39] R. Chandra, R. Mahajan, V. Padmanabhan, and M. Zhang. *CRAWDAD data set microsoft/osdi2006*, May 2007. http://crawdad.cs.dartmouth.edu/microsoft/osdi2006.

[40] Caleb Phillips and Suresh Singh. *CRAWDAD data set pdx/vwave*, Sept. 2007. http://crawdad.cs.dartmouth.edu/pdx/vwave.

[41] Xiofei Wang. *CRAWDAD trace set snu/wow_via_wimax*, Oct. 2009. http://crawdad.cs.dartmouth.edu/snu/wow_via_wimax.