DOF: A Local Wireless Information Plane

Steven Hong, Sachin Katti
Stanford University
{hsiying,skatti}@stanford.edu

Abstract

The ability to detect what unlicensed radios are operating in a neighborhood, their spectrum occupancies and the spatial directions their signals are traversing is a fundamental primitive needed by many applications, ranging from smart radios to coexistence to network management to security. In this paper we present DOF, a detector that in a single framework accurately estimates all three parameters. DOF builds on the insight that in most wireless protocols, there are hidden repeating patterns in the signals that can be used to construct unique signatures, and accurately estimate signal types and their spectral and spatial parameters. We show via experimental evaluation in an indoor testbed that DOF is *robust and accurate*, it achieves greater than 85% accuracy even when the SNRs of the detected signals are as low as 0 dB, and even when there are multiple interfering signals present. To demonstrate the benefits of DOF, we design and implement a preliminary prototype of a smart radio that operates on top of DOF, and show experimentally that it provides a 80% increase in throughput over Jello, the best known prior implementation, while causing less than 10% performance drop for co-existing WiFi and Zigbee radios.

Categories and Subject Descriptors

C.4 [**Computer Systems Organization**]: Performance of Systems

General Terms

Algorithms, Performance, Design

1. INTRODUCTION

The ability to detect what unlicensed radios are operating in a neighborhood, what parts of the spectrum they are occupying, and what spatial directions their signals are traversing is a fundamental primitive that is needed by many applications. For example, smart and agile radios such as [28, 22] could use it to detect what spectral resources are unused, and exploit them to provide high throughput. They could detect what spatial directions are unoccupied, and directionally steer their signals to further increase capacity. They could also use the primitive to be gentle when needed, if a low power medical wireless sensor is operating in the neighborhood, the smart radio could detect it and take extra measures to avoid causing interference to the sensor, lest some critical communication is impaired. Similarly, network administrators can use such a primitive to manage their "airspace", improve channel allocation and diagnose performance problems. Recent work [27] has explored using detectors that compute what spatial directions signals arrive at for wireless network security. Thus, a large and growing number of applications could benefit from such a primitive.

However, building such a detector that operates accurately across the large range of SNRs signals exhibit, in the presence of multiple interfering signals, or in the rich indoor multipath environment of the unlicensed ISM band is hard. Prior implemented systems have mostly focused on spectrum occupancy detection, and used threshold based methods that estimate changes in received signal energy [16] or the variations in the FFT [28] to estimate spectrum occupancy. However, optimal thresholds that work accurately across the rich variety of conditions (in SNR, multipath, interference etc) are hard to pick, and consequently these methods have low accuracy. Other work [22, 16] has used higher layer protocol behavior signatures to detect radio types. However, these techniques also rely on threshold based methods to detect the protocol behavior, and suffer from the same problems as above.

In this paper we present **D**egrees **O**f **F**reedom (DOF), *a single framework that accurately detects what radios exist in a neighborhood, what parts of the spectrum they occupy, and their angles of arrival (AoA)* at the detector. We believe this to be a first. DOF is *robust* and works accurately (around 90% accuracy) in a large SNR range (0 to 30dB) as well as in the presence of multiple interfering signals. DOF is *passive* and does not impose any measurement overhead, it can operate even when the detecting radio is being used for other communication. Finally, DOF is *efficient* to implement, it builds on top of commonly available FFT modules and requires modest extra resources (30% more computation compared to a standard FFT).

The key insight behind DOF is the observation that for most wireless protocols, *there are hidden repeating patterns that are unique and necessary for their operation*. For example, Wifi uses a repeating cyclic prefix to avoid intersymbol interference between consecutive OFDM symbols. A Zigbee radio has a repeating pulse which it uses for QPSK data transmission, Bluetooth has a Gaussian pulse on which it modulates data bits using FSK that is repeating with a different frequency and so on. DOF exploits the existence of these patterns to create unique signatures for each signal type. Further, DOF shows that the same signatures can also be exploited to determine the spectrum occupied and the AoA of that signal type.

Algorithmically, DOF extracts feature vectors using the following key idea: if a signal has a repeating hidden pattern, then a delayed version of the signal correlated with the original signal will show peaks at specific delay intervals. These intervals form a signature for each signal and can be used to extract feature vectors. We build on

SIGCOMM'11, August 15–19, 2011, Toronto, Ontario, Canada.

prior work [9, 19] in cyclostationary signal analysis to design an efficient feature extraction technique based on standard FFT operations. However, DOF's key contribution over prior work in cyclostationary analysis is to show that the extracted feature vector encodes information about the component signal types, what spectrum they occupy, as well as what AoAs they arrive at the detecting radio. DOF designs a novel SVM decision tree to classify component signal types, and new algorithms to estimate their spectrum occupancies as well as AoAs from the feature vector.

We implement DOF using the fftw [1] library and GnuRadio [3] software on a wideband radio that is capable of operating over the entire 100 MHz ISM band and has 4 MIMO antennas. We evaluate DOF using testbed experiments in an indoor office environment and compare it to three prior approaches, RFDump [16] for signal type, Jello [28] for spectrum occupancy, and SecureAngle [27] for AoA estimation (the best known implemented systems for each component respectively). We find that:

- DOF is accurate and robust at all SNRs, it classifies co-existing radio types with greater than 85% accuracy even at SNRs as low as 0dB. On the other hand, RFDump is at most 60% accurate at SNRs lower than 8dB.
- DOF is robust to interference, achieving more than 82% accuracy in detecting component signal types even when there are three overlapping and interfering signals. The compared approach RFDump cannot operate in this case.
- DOF's spectrum occupancy estimates are more than 85% accurate at low SNRs or in the presence of interference. The compared approach, Jello has an error of 35%, and cannot detect individual spectrum occupancies of interfering component signals.
- DOF's AoA estimation error is less than 6 degrees for SNRs as low as 0dB, and is the same as SecureAngle.

DOF is practical and can be applied to many problems. While we leave most of DOF's applications to future work, we demonstrate the potential benefits of DOF for building smart and agile radios by designing and implementing a preliminary prototype, DOF-SR. The key novel component in DOF-SR is that it's aggressiveness in scavenging for unused spectral resources can be tuned by a user specified policy so that interference to co-existing radios is controlled. To demonstrate this flexibility we implement three sample policies, from one which only uses unoccupied spectrum and minimizes interference to co-existing radios to ones which use microwave oven occupied spectrum and compete with co-existing WiFi radios. We deploy DOF-SR in our indoor testbed and compare it with Jello [28] (which uses edge detection for finding unused spectrum). Our evaluation shows that DOF-SR provides nearly a 80% throughput increase over Jello in crowded environments. Further, the co-existing WiFi/Zigbee radios suffer less than 10% throughput drop with DOF-SR, while Jello can cause nearly a 45% throughput drop. DOF-SR outperforms because it can accurately detect (un)occupied spectrum even at low SNRs as well as the occupying signal types, allowing it to more accurately scavenge unused spectrum, yet guarantee that it does not affect the co-existing radios.

2. RELATED WORK

DOF bridges and builds upon related work in signal detection and cyclostationary signal analysis. We discuss both of them below.

2.1 Signal Detection

Detecting Radio Type: Prior work such as RFDump and others [16, 22] has used unique protocol characteristics (e.g. 10μs delay between data and ACK WiFi packets) to infer radio type. The basic approach is to detect the start and end of packets using energy detection in the time domain, and use the delays between packets to estimate radio type. However, energy detection is not accurate at medium to low SNR, and fails if there are multiple interfering signals as we show in our evaluation in Sec. 6. Other work [20] has used preamble correlation to detect radio type by exploiting known preambles at the start of a packet. However this technique doesn't work for legacy analog signals such as microwaves, cordless phones etc which don't have preambles. Further, as prior work has shown [10], preamble correlation requires coarse synchronization to the carrier frequency of the detected signal, which becomes expensive given the large number of carrier frequencies for different radio types in the ISM band.

Detecting Spectrum Occupancy: Prior work such as Jello [28] has used edge detection on the power spectral density of the received signal to estimate spectrum occupancy. The basic idea is to compute the slope of the PSD at every point, and detect signal starts and ends based on thresholds on the slope. However, at low SNRs and for signals whose spectral masks are not of good quality, the accuracy of this approach is low because noise and spectral leakage can cause sharp spikes in the slope away from where the signal is located. Further, this approach fails when we have multiple interfering signals who also overlap in the frequency domain, since a edge will be detected as soon as the first signal ends, in spite of the second signal which occupies some more portion of the spectrum. Other approaches based on energy detection such as SpecNet [12] also suffer at low SNRs and are unable to distinguish between overlapping signals.

Detecting Angle of Arrival: Prior work such as SecureAngle [27] has used classic AoA estimation algorithms [14, 6, 23] to compute AoAs of the incoming signals. These approaches are highly accurate, and we show in our evaluation that DOF's accuracy is similar. Further, DOF can automatically associate a signal type with the AoA (e.g. a WiFi signal is impinging at 45^o), while prior approaches need separate detectors to associate signal type.

DOF thus provides a single framework that estimates all three parameters, and with accuracy better than the best known implemented techniques for each component.

2.2 Cyclostationary Signal Analysis

DOF builds on prior work in cyclostationary signal analysis, which was pioneered in the early 90's through the work of Gardner [9], and has been used widely in a variety of applications [11, 24, 17, 29]. Further, recent work [8, 7] has used neural network classifiers with cyclostationary features to detect the type of modulation used in a received signal. Finally, recent work has implemented cyclostationary techniques on the USRP platform [21, 19, 4] and evaluated its effectiveness for detection and rendezvous in cognitive networks.

As we will see in Sec. 3, DOF builds on this prior work to design an efficient feature extraction technique. However, DOF differentiates itself from all prior work in cyclostationary signal analysis in the following ways:

- DOF designs an efficient linear-time classification technique based on hiearchical SVMs to estimate the type of multiple overlapping signals. Prior approaches based on neural networks [8, 7] have cubic computational complexity and those based on SVMs [15] are limited to classifying a single signal. DOF's technique is robust to the presence of multiple interfering signals and can reuse the same SVM decision tree for classifying all component signal types. To the best of our knowledge, we are not aware of prior work in cyclostationary analysis that has handled detection of multiple interfering signals.
- DOF extends cyclostationary signal analysis to detect angle of arrivals, and designs a novel algorithm that computes AoAs as well as associates the signal type with the signal on each AoA.
- DOF is implemented on a wideband radio, and has been eval-

uated extensively in an indoor testbed with five different interfering signal types (WiFi, Bluetooth, Zigbee, Analog Cordless phones and microwave signals). We are not aware of any work that provides a similar extensive evaluation.

- We also design and build a preliminary prototype of a smart radio based on DOF, and show experimentally how it can be used to increase network capacity without harming other radios.

3. OVERVIEW & DESIGN

DOF operates on windows of raw samples from the ADC which do not undergo any demodulation, decoding or synchronization. These raw samples are processed to extract feature vectors, which are then used to detect signal types, the corresponding spectrum occupancies and the AoAs of the signals at the detector. Before discussing the detailed design, we provide the high level intuition behind DOF.

3.1 Intuition

The key insight behind DOF is that almost every radio protocol used for communication has hidden repeating patterns. For example, an OFDM PHY (used in WiFi) has a cyclic prefix (CP) where at the end of each OFDM symbol block, the symbols from the start are repeated. The CP serves two purposes, first it helps in avoiding inter-symbol interference, and second it helps in preserving orthogonality of the OFDM subcarriers [26]. Thus a CP is an important attribute of the OFDM PHY itself, and necessary for its correct operation. Similarly, every other protocol operating in the ISM band has repeating patterns, that are unique and needed for their correct operation.

Note that these patterns are fundamental to the corresponding physical layers and are present in every packet (data, ACK and for every bitrate). These patterns are not some quirk of a specific hardware implementation or PHY layer parameter setting (e.g. different channel transmission times for a 1500B packet based on what bitrate is used in WiFi). Hence these patterns can potentially form a robust signature that is invariant to differences in hardware or PHY layer parameters.

How can we use the existence of these hidden patterns to detect the signal type, occupied spectrum and angle of arrival? We can use the following key trick from cyclostationary signal analysis [9]: *if a signal has a repeating pattern, then if we correlate the received signal against itself delayed by a fixed amount, the correlation will peak when the delay is equal to the period at which the pattern repeats*. Specifically, lets denote the raw signal samples we are receiving by $x[n]$. Consider the following function

$$R_x^\alpha(\tau) = \sum_{n=-\infty}^{\infty} x[n][x^*[n-\tau]]e^{-j2\pi\alpha n} \quad (1)$$

For an appropriate value of τ corresponding to the time period between the repeating patterns, the above value will be maximized, since the random patterns in $x[n]$ will be aligned. Further, these peak values occur only at periodic intervals in n. Hence the second exponential term $e^{-j2\pi\alpha n}$ is in effect computing the frequency α at which this hidden pattern repeats. We define such a frequency as a *pattern frequency*, and Eq. 1 is known as the Cyclic Autocorrelation Function (CAF) [9] at a particular pattern frequency α and delay τ. The CAF will exhibit a high value only for delays and pattern frequencies that correspond to repeating patterns in the signal.

Figure. 1 shows the 2-D CAF plots for a received signal that has WiFi and Zigbee signals interfering with each other. As explained above, WiFi uses OFDM, and has a repeating cyclic prefix, as well as other repeating patterns. In the CAF plot, we see spikes corresponding to these repeating patterns at different pattern frequencies and delays. Similarly, the Zigbee signal shows spikes at pattern frequencies corresponding to how its pulse repeats. Note the stark difference

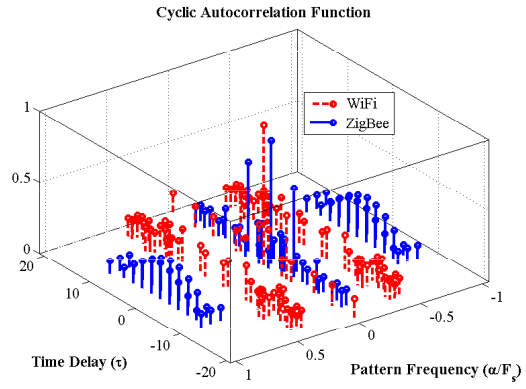

Figure 1: Cyclic Autocorrelation Function for WiFi and Zigbee - The spikes at different pattern frequencies are due to the repeating OFDM CP, and the repeating pulse on which QPSK symbols are modulated in Zigbee.

in the locations of the spikes for pattern frequencies for WiFi and Zigbee. The differentiability in spike locations enables DOF to distinguish both signals even when they are interfering with each other.

DOF uses the locations of these pattern frequencies as signatures for different signal types. In the following sections we expand on this insight and explain the design of the classifier, spectrum occupancy and AoA detection algorithms, which are DOF's main and novel contributions. However, to make these algorithms practical, we first need to efficiently evaluate the Cyclic Autocorrelation Function at the relevant pattern frequencies. Hence we first discuss DOF's feature extraction step, which borrows ideas from cyclostationary signal processing to design an efficient extraction algorithm.

4. DESIGN

DOF's design consists of 4 stages and an overview of the architecture is shown in Figure 2.

4.1 Feature Extraction

DOF's feature extraction component computes feature vectors from the digital samples delivered by the ADC. Our algorithm builds on a rich body of prior work in cyclostationary signal analysis [9], and is conceptually similar to recent work in whitespace radios that uses cyclostationary analysis to detect primary TV transmitters. Our main contribution here is the adaptation of the algorithm to work for the multitude of signals in the ISM band and an efficient implementation that works on a 100MHz wideband radio.

As described in 3.1 the feature extraction step is supposed to find the prominent pattern frequencies which represent the frequencies at which repeating patterns manifest in the different PHYs. However, instead of using the CAF defined in Eq. 1, we use an equivalent representation called the Spectral Correlation Function (SCF) [9]:

$$S_x^\alpha(f) \quad = \sum_{\tau=-\infty}^{\infty} R_x^\alpha(\tau)e^{-j2\pi\tau} \quad (2)$$

The SCF is equal to the frequency transform of the CAF. Since frequency transforms are unitary, both representations are equivalent. If the CAF peaked for a certain value of τ, then the SCF will peak for a particular value of f that is inversely proportional to τ. Intuitively, the reason for this is that if a hidden pattern repeats at a lag of τ, then by definition it repeats for every integer multiple of τ.

The reason for moving to the SCF is that it can be computed efficiently [21] for discrete time windows as follows

$$S_x^\alpha(f) = \frac{1}{L}\sum_{l=0}^{L-1} X_{lN}(f)X_{lN}^*(f-\alpha) \quad (3)$$

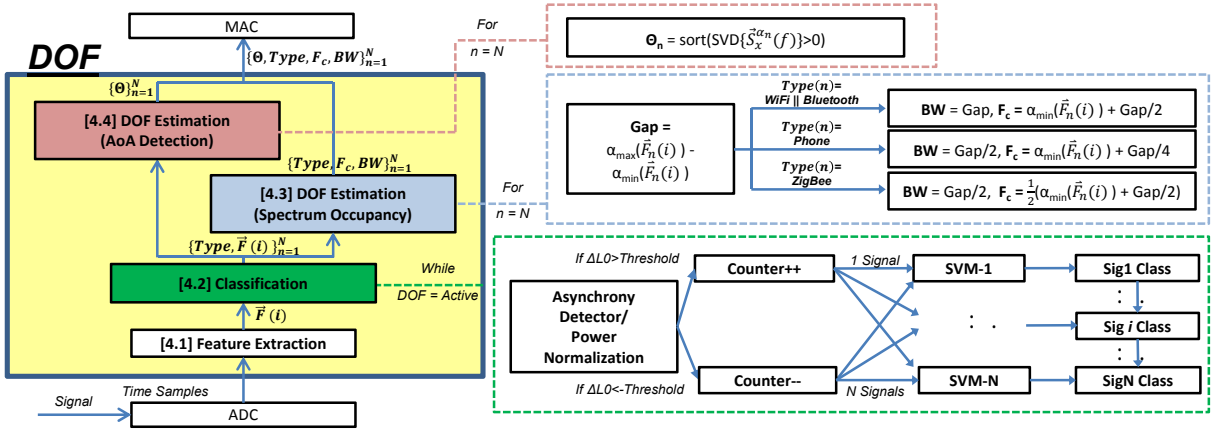

Figure 2: Overview of DOF showing the overall architecture and where it sits in the stack. Extracted features are first classified by signal type and then processed to determine which wireless degrees of freedom are in use. DOF then passes the distilled information up to the MAC layer which can utilize the information as it sees fit.

where $X_{lN}(f)$ is the FFT of the received signal for the l'th time window of length N samples, $*$ is the complex conjugate, and the summation is over L consecutive time windows of the received signal.

The key thing to note in Eq 3 is that the SCF can be expressed as a product of the FFTs of the received signal. Hence to compute the SCF at any pattern frequency α, one just has to take the product of the received signal's FFT with itself albeit shifted in the frequency domain by α. FFTs are very efficient to implement in hardware [13], and any wireless PHY that would use OFDM would already have an FFT hardware module. Hence we believe that the SCF can be easily computed using existing hardware. We compute and evaluate the computational complexity and verify the above claim in Sec. 5.

Feature Extraction: Finally, we summarize DOF's feature vector. Given the universe of signal types to detect (WiFi, Zigbee, cordless phones, microwaves and Bluetooth currently), we first determine the union of the unique sets of pattern frequency and frequency tuples contained in each type's signature. Let this union consist of the following M tuples, (α_1, f_1), ..., (α_M, f_M), then the feature vector $\vec{F}$ is defined as:

$$F(i) = (S_x^{\alpha_i}(f_i)) \qquad \forall i = 1, \ldots, M \tag{4}$$

The components of the feature vector are values of the SCF at different points, unique to the corresponding signal types.

4.2 Estimating Signal Type

DOF designs a novel decision tree based on SVMs [5] which allows it to classify multiple component signal types in an interfered signal using the extracted feature vectors. A SVM classifier takes an input feature vector, $\vec{F}$, and predicts the signal type T if any that exists in the received signal. These classifiers are trained using a small labeled dataset. It's common to regularize the feature vectors using a kernel function such as a Gaussian kernel [5] and project them to higher dimensions to make the feature vectors belonging to different types linearly separable, we use the same technique in DOF.

A naive method of using these classifiers is to train a SVM classifier using labeled data collected by transmitting from a particular radio and computing the corresponding feature vector from the received signal, and doing so for different radio types and locations. However, this generic off-the-shelf SVM design fails to work. The reason is that DOF expects to accurately detect signal types even when the received signal has multiple interfering signals in it. Interference significantly distorts feature vectors and throws off the SVM classifier. Specifically, the SCF for an interfered signal at a particular pattern

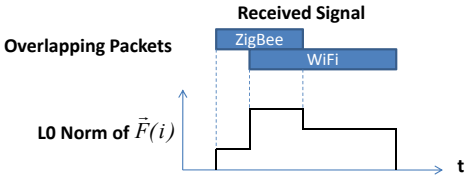

Figure 3: Detecting the Number of Signals - Asynchrony between packets causes differences in the L0 Norm of $\vec{F}$

frequency α can be shown to be equal to [9]:

$$S_X^{\alpha}(f) = a^2 S_{X_1}^{\alpha}(f) + b^2 S_{X_2}^{\alpha}(f) + R \tag{5}$$

where X_1 and X_2 are the interfering signals with amplitudes a and b. R is a residual term representing cross-talk between the two signals. Thus the feature vector will be a sum of the feature vectors if the signals alone had been present without interference scaled according to their respective powers, plus a term that represents the crosstalk. The unique pattern frequencies for each component signal type are retained, but after kernel regularization, the test feature vector itself will not correspond to any of the training feature vectors the SVM classifier has been trained on.

One naive approach to this problem would be to train SVMs for all possible combinations of signals. However, this approach quickly gets out of hand, since the classifier has to account for the fact that the interfering signals will have different unknown powers, and consequently the feature vectors will be clustered differently for each combination of powers. Training classifiers for all possible signal combinations and powers is prohibitively expensive.

4.2.1 *Robust & Efficient Classification*

DOF builds a decision tree that can efficiently identify multiple component signal types in an interfering signal via two steps:

1) Exploiting Asynchrony: Transmissions from different nodes in the real world rarely overlap with each other perfectly since transmissions from two independent nodes will very likely be asynchronous as shown in Fig. 3. DOF exploits this idea to compute two quantities: the number of component signals in the received signal, and their average individual power.

To determine how many signals are present, DOF uses the following idea: if a new signal starts interfering, then the feature vector DOF extracts will start showing many new non-zero components due

to the unique features belonging to the new signal. Hence, we can use the following algorithm to compute the number of interfering signals:

1. Keep track of the l_0 norm (i.e. the number of non-zero components) of the computed feature vector.
2. If the l_0 norm exhibits a sudden shift, then declare a change in the number of interfering signals. If the l_0 norm shift is positive, then a new signal has started interfering, if the change is negative, then one of the interfering signals has stopped.

The above algorithm begins by initializing the counter for the number of signals to zero. Hence, the algorithm continuously keeps track of the number of interfering signals at any point.

Second, DOF exploits the fact that the total power of the received signal is equal to the sum of the powers of the constituent signals and noise. Hence, as the received signal samples are received, DOF keeps a moving window average of the power at that point. If DOF detect a new signal, it estimates the power of the new signal, as the new received signal power minus the received signal power before the presence of a new signal was detected. Thus, DOF detects the number of component signals, as well as their powers in the received signal. It exploits this information in classifying the constituent signal types, as we explain next.

2) Constructing the SVMs: DOF exploits knowledge of the number of signal types and their powers computed above to design an efficient SVM decision tree for classification. The basic idea is to train a small number of classifiers equal to the number of signal types we wish to detect (currently five signal types in our implementation): one classifier for the case where the received signal has zero or one signal type, another classifier when the received signal has two signal types and so on. These classifiers are trained with labeled datasets that are generated by taking labeled data from experiments where there is a single signal type in the collected data, and adding them up after normalizing their powers. For example, if we have labeled data containing WiFi signals at power P_1 and another labeled dataset containing Zigbee signals at power P_2, to create one labeled data point for the classifier meant for two signals, we would add the two datasets above after normalizing their powers to be equal. By taking different numbers and combinations of signal types and repeating the above procedure, we create five training sets for the five SVM classifiers.

The above technique has two advantages. First, we only need to train five classifiers, significantly smaller than the naive approach which needs at least 31 different SVM classifiers (one for each combination of signal types and possibly more for different powers). Second, collecting training data is relatively easy, since we only have to collect data from controlled experiments where there is a single radio operating, and we can artificially add them up later to generate data for classifiers attempting to detect multiple interfering signal types.

To use these classifiers in practice however, we need to normalize the amplitudes of the computed feature vectors since the classifiers were trained on data where the component signals had equal power. To accomplish this, we exploit that we can compute the powers of the individual signals using asynchrony as we explained in the previous section. For example, lets say we are classifying a signal which we have estimated to have two different component signals with powers P_1 and P_2, and the signal X_1 starts before X_2. Due to asynchrony, we have an interference free part of X_1 and consequently an interference free estimate of the corresponding feature vector F_1. When we get to the part of the signal where these two signals interfere, we multiply the components of the new feature vector that were also non-zero in the original feature vector by P_2, and the remaining components by P_1. In effect, we have normalized the feature vectors corresponding to both components to have the same amplitude P_1P_2. Now, the classifiers that were trained on normalized data can proceed to classify the component signal types.

The above technique recursively generalizes to any number of interfering signals, since we can use the above procedure whenever we detect that a new signal has started interfering. Similarly, we can reverse the technique when we detect that one of the signals has stopped. Specifically, if we detect via the l_0 norm technique that the number of signal types has reduced by 1, and the total observed power drops by P', then we just normalize the remaining feature vector components by $1/P'$.

4.3 Estimating Spectrum Occupancy

After identifying signal type, DOF computes the carrier frequency and bandwidth of each signal type. The key idea is that the feature vectors that were extracted for detecting type also encode information about the carrier frequency and the bandwidth of the signal. The reason is that almost every wireless communication signal modulates constellation symbols (e.g. QAM) on top of standard bandwidth-limited pulses such as raised cosine filters. The pulse rate is directly proportional to the bandwidth for that signal (e.g. 5MHz for Zigbee). This repeating pulse gives rise to specific pattern frequencies whose value is a function of the bandwidth and the carrier frequency of that signal. For OFDM signals like WiFi, instead of a pulse we have the CP that repeats at a frequency proportional to the bandwidth of the signal. DOF leverages these relationships in building its spectrum occupancy estimation algorithm.

To see why feature vectors encode information about the carrier frequency and bandwidth, consider the following BPSK signal that is representative of transmitted wireless signals

$$s(t) = b\cos(2\pi f_b t)e^{j2\pi f_c t} \tag{6}$$

where $b = \pm 1$ represents the bits and the $\cos(2\pi f_b t)$ represents the pulse on which the bits are modulated, and f_b is the bandwidth used for transmission, and f_c is the carrier frequency. Note that typically for spectrum masking purposes more specialized pulses than simple cosines are used, but for our explanation, this representation suffices.

Lets assume the center frequency of our detector is $f_{c'}$ and the gap with the transmitted signal's carrier frequency is $\delta f = |f_c - f_{c'}|$. This gap just shifts the FFT of the signal by the same amount δf. To see how the SCF for the received signal looks, lets first compute the CAF for this with $\tau = 0$

$$CAF(s(t)) = b^2 e^{-2\pi\delta f t} + b^2 cos(4\pi f_b t) \tag{7}$$

As discussed before, the SCF is just the FFT of the CAF. From the above equation it becomes clear that when we take its FFT, we will see two spikes, one at δf, and one at $2f_b$, giving us two prominent pattern frequencies at these locations. The location of the two pattern frequencies along with the knowledge of the detector's center frequency $f_{c'}$ is sufficient to compute the bandwidth and carrier frequency of the transmitted signal.

The above technique generalizes to every communication radio (including analog radios such as cordless phones), i.e. the Spectral Correlation Function of a signal will exhibit a prominent value at a pattern frequency corresponding to some function of f_c, f_b. Table 1 lists the pattern frequencies that are observed in the SCF which are direct functions of the carrier frequency and occupied bandwidth for different signal types. This table serves as the basis of DOF's algorithm for spectrum occupancy and carrier frequency estimation.

However, the above technique has two caveats. First, for Bluetooth signals which employ frequency hopping over 1 MHz intervals at a rate of 1600 hops/second, the per hop period is $1/1600 = 625\mu s$. In our current implementation, our spectrum occupancy algorithm runs over a window of roughly $1ms$ intervals. Hence, DOF may estimate multiple spectrum occupancies for Bluetooth signals, since a Bluetooth signal could hop multiple times in 1ms. Second, the above

Table 1: Relationship between Pattern Frequencies and Bandwidth/Carrier Frequency

Signal Type	Pattern Frequency Locations
WiFi	all α's between $[f_c - \frac{BW}{2}, f_c + \frac{BW}{2}]$
Bluetooth	$f_c, f_c + \frac{BW}{2}, f_c - \frac{BW}{2}$
Analog Phone	$f_c, f_c + BW, f_c - BW$
ZigBee	$2f_c + BW, 2f_c - BW$

intuition does not work for non communication signals such as microwave ovens because we are unable to exploit packet asynchrony to determine the number of signals present in the time window. However, as prior work has shown [25], microwave signals can be modeled as FM signals with a sweeping bandwidth that is equal to the AC power switching frequency. We can leverage this model to initialize our asynchrony detector counter based on the number of feature vectors when the counter is set to zero, and also compute the occupied spectrum for microwave signals due to these feature vectors.

Finally, note that one cannot determine bandwidth occupancy directly from signal type. While the detected signal type (e.g. WiFi) can tell us what is the expected signal bandwidth (e.g. 20MHz for Wifi), it cannot tell us what carrier frequency is used since WiFi has 11 different channels. The above technique determines both bandwidth and carrier frequency directly from the feature vectors.

4.4 Estimating Angles of Arrival

The final component of DOF is angle of arrival (AoA) estimation for each signal type detected. DOF designs a novel and efficient algorithm that extends cyclostationary analysis to also compute AoAs. The key insight is that we can leverage already known information about the unique pattern frequencies corresponding to a signal type to extract their AoAs. We demonstrate the basic idea using a simple uniform linear MIMO antenna array (ULA) [23] as our antenna geometry. Our algorithm generalizes to any antenna geometry, but ULA suffices for exposition.

Lets assume that we have M antennas and our radio receives $N < M$ signals that exhibit pattern features at unique $\alpha_n \; \forall n = 1, \ldots, N$ and arrive at AoAs $\theta_i \; \forall i = 1, \ldots, N$ respectively. A uniform linear array by definition has all its antennas on a line with equal spacing between them as shown in Fig. 4. Because the antennas are equally spaced, a signal at a particular angle of arrival θ has a difference in propagation distance that results in a time delay at the m^{th} antenna with respect to the first antenna of

$$\tau_m(\theta) = (m-1)\frac{d \sin\theta}{c} \tag{8}$$

where c is the rate of propagation (speed of light for free space) through the medium and d is the inter-antenna spacing. A delay in the time domain manifests itself as a phase shift as long as the narrowband assumption holds (the bandwidth of the signal does not exceed the channel's coherence bandwidth) and so the received signal at the M antennas modeled as a summation of all the interfering components is equal to

$$\begin{aligned} \boldsymbol{y}(t) = & \sum_{n=1}^{N} \phi(\theta_n) x_n(t) + \boldsymbol{n}(t) \\ = & \quad \boldsymbol{\Phi x}(t) + \boldsymbol{n}(t) \end{aligned} \tag{9}$$

where $\boldsymbol{x} = [x_1 \ldots x_N]^T$ with each x_n corresponding to the signal arriving at angle θ_n, $\boldsymbol{y} = [y_1 \ldots y_N]^T$ is the vector consisting of signals received at the M antennas, $\phi(\theta_n) = [1 \; e^{j2\pi f_c \tau_2(\theta_n)} \ldots e^{j2\pi f_c \tau_M(\theta_n)}]^T$ and $\boldsymbol{\Phi} = [\phi(\theta_1) \; \ldots \; \phi(\theta_n)]$ where f_c is the carrier frequency.

The objective of any AoA estimation algorithm is to compute the N column vectors in the AoA matrix $\boldsymbol{\phi}$, since they directly provide the corresponding AoAs for each of the N signal types. The typical approach is to do a search over the space of possible matrices, and algorithms differ in how the search is conducted. The key contribution in our algorithm is a way to leverage the computed pattern frequencies to significantly reduce the search space and thus enable fast AoA computation, as well as automatically associate the computed AoAs with the corresponding signal type.

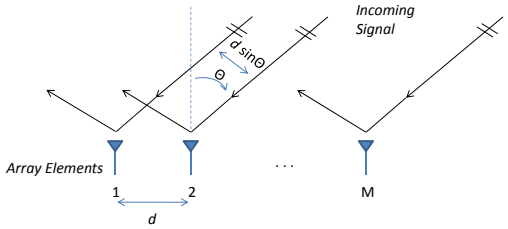

Figure 4: Uniform Linear Array - Sensing a plane wave impinging at an angle θ

Lets assume we have detected a particular signal type and that it has a unique *pattern frequency* α_u and is arriving at a single AoA θ_u. DOF's algorithm first computes the Spectral Correlation Matrix $\mathbf{S_y^{\alpha_u}}$ of the received signal at the M antennas [1] at the unique pattern frequency α_u. We omit the proof for brevity, but we can show that this matrix is related to the AoA vector $\phi(\theta_u)$ as follows:

$$\vec{S}_y^{\alpha_u}(f) = \Phi(\theta_u)\vec{S}_x^{\alpha_u}(f)\Phi(\theta_u)^H \tag{10}$$

where $(\cdot)^H$ denotes the conjugate transpose operation. Since α_u is unique to this signal type, $\vec{S}_x^{\alpha_u}(f)$ will be a diagonal matrix, which implies that $\Phi(\theta_u)$ is the eigenvector of the computed matrix $\vec{S}_y^{\alpha_u}$. Hence, in order to compute the AoA for this signal type, we just have to compute the eigenvector of the matrix computed in Eq. 10. Because this computation is only performed at the *pattern frequencies corresponding to the received signal* and not all possible pattern frequencies, we are able to reduce the overall computation and associate signal type with each angle.

In practice due to multipath effects, each signal type will arrive at multiple AoAs. Due to this instead of a single eigenvector as above, we will have multiple eigenvectors, each corresponding to a different angle at which this signal arrives.

There are two important takeaways from this section:

- By detecting signal types we obtain a list of corresponding unique pattern frequencies. These are directly used in the AoA algorithm described above to efficiently calculate AoAs.
- By the very nature of the algorithm, i.e. our use of the unique pattern frequencies for the detected signal types, the computed AoAs are naturally and accurately associated with the corresponding signal types.

5. IMPLEMENTATION

DOF is implemented in C using a fast FFT implementation from FFTW [1] on a PC with an Intel Core i7 980x processor and 8GB of RAM. We use a wideband radio [18] (shown in Fig. 5) with a frontend bandwidth of 100MHz spanning the entire ISM band. The wideband radio is a modified channel sounder that was originally designed for taking channel measurements by sending user specified pilots. We modify the frontend to be able to send and receive arbitrary waveforms in the entire 100MHz ISM band. The frontend has a carrier

[1] a generalization for MIMO signals of the Spectral Correlation Function defined in earlier sections for single signals

20MHz	40MHz	60MHz	80MHz	100MHz
0.4	0.8	1.4	1.8	2.5

Table 2: Microbenchmarks - CPU time normalized wrt actual signal time of the trace

frequency of 2.45GHz and a max output power of 15dBm. However, similar to other SDR platforms such as USRP2s, the interconnect between the SDR frontend and the PC does not meet the latency requirements needed to implement timing sensitive MAC functions such as ACKs. DOF's algorithms operate on the raw digital samples collected by the wideband frontend. We provide a microbenchmark for our implementation in Sec. 5.1.

5.1 Complexity

In this section we discuss the computational complexity of DOF. We compare DOF's complexity against the simple and widely used PSD based edge/energy detector [28, 16, 22]. For AoA estimation, we compare it against the MUSIC algorithm [23] used in prior work such as SecureAngle [27].

Computational Complexity: The main computationally intensive task in DOF is the feature extraction step, which involves computing Eq. 3 for every component in the feature vector. The complexity is dictated by the choice of the FFT length N and the averaging window L. Higher values of N and L provide better resolution for the FFT and SCF respectively [9] and consequently higher accuracy for DOF, but also increase complexity. In our current implementation, we find that $N = 512$ and $L = 16$ suffices for DOF to work accurately over the 100MHz ISM band. Prior energy/edge based approaches [28] use a 256 point FFT, but were implemented over narrowband USRP2 radios with at most 10MHz bandwidth, while DOF works over a wideband radio with 100MHz bandwidth. We believe that prior work would have to use at least a 512 length FFT to operate over such widebands, otherwise the spectral resolution would be too low resulting in inaccuracy. (we verified the inaccuracy with 256 length FFTs experimentally for one prior approach [28]).

DOF and edge/energy detection share the same FFT complexity [2] of $5N \log N = 20384$ floating point operations per window. Next, DOF computes the $K = 80$ feature vector components by averaging over 16 windows, which costs another $4 * 16 * 80 = 5120$ floating point operations. Note that prior FFT based approaches [28, 22] also have to perform this averaging to smooth the FFT and avoid false positives. Our radio type classifier has an l_0 norm estimator and equalizer, that require $K + N \approx 600$ comparisons. The SVM classifiers require $K = 80$ real multiplications, while the spectrum occupancy estimation algorithm requires a small number of extra operations equal to the number of signal types detected. Thus in total, DOF requires 6000 extra floating point operations, in addition to the 20384 floating point operations that the FFT requires. Hence DOF's extra complexity is less than 30% over a standard FFT which we believe is reasonably modest.

Energy detection of course cannot compute AoAs, hence we compare DOF's complexity with the MUSIC algorithm [23] that is used in prior work [27]. The order computational complexity of MUSIC as well as DOF's AoA estimation algorithm is $O(PM^3)$, where M is the number of antennas and P is the number of distinct AoAs. However, we find empirically that the constant in the order notation is significantly smaller for DOF. This is because the MUSIC algorithm involves computing eigenvectors for a series of matrices as it converges to the correct AoAs. DOF on the other hand has to compute the eigenvectors only once as described in Section 4.4.

To summarize, DOF has modestly higher if not similar computational complexity compared to traditional energy/edge detectors, and actually lower complexity than other AoA methods. However, as we will see in the next section, DOF significantly outperforms energy/edge based approaches and has additional features such as signal type detection that energy/edge based detectors do not provide. Hence, we believe that the additional complexity is a reasonable tradeoff given the significant gains in functionality and accuracy.

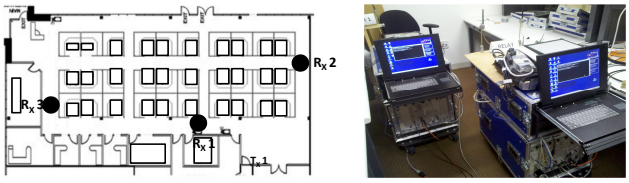

Figure 5: Testbed layout and wideband software radio

Micro-benchmark: Table 2 provides benchmark results for DOF's current software implementation. We calculate the normalized time by dividing the wall clock time used by our system divided by the actual signal time on the air. The goal is to see how close to "realtime" our system is. We provide benchmarks as we vary the bandwidth of the radio from 20MHz to 100MHz in increments of 20MHz. A larger bandwidth naturally means a faster stream of data to keep up with.

DOF performs in realtime for radios with bandwidths of up to 40MHz and starts falling behind with higher bandwidths. However, this is a software based implementation of the FFT (which requires the most computation), and we believe a hardware implementation would be significantly faster and be able to handle higher bandwidths. Further these benchmarks compare favorably with prior work [16].

6. EVALUATION

In this section we evaluate the accuracy of DOF and determine how different factors such as signal SNR, the number of interfering signals impact its performance using testbed experiments. Our current implementation is geared towards 5 common signal types in the ISM band - WiFi, Zigbee, Bluetooth, analog/digital cordless phones and microwave signals.

We first summarize our findings:

- DOF's performance is robust to the SNR of the detected signals. We find that DOF achieves greater than 85% accuracy even when the SNR of the detected signals is as low as 0dB. The best known prior approach have errors greater than 40% for SNRs below 8dB. [16]
- DOF's performance is robust to interference between detected signals. We find that DOF accurately classifies all component signals with greater than 82% accuracy even with 3 interfering signals. Prior approaches do not work with interfered signals.
- DOF's spectrum occupancy estimates are at least 85% accurate, at SNRs as low as 0dB and in the presence of multiple overlapping and interfering signals. The best known prior approach achieves an accuracy of 65% under similar conditions.
- DOF's AoA estimation is as accurate as the best known prior technique [23]. Further, unlike prior work it accurately associates the estimated AoAs with the correct type for the signal arriving at that angle.

Compared Approaches: We compare against the best known implemented systems for each component in DOF. First we compare against RFDump [16] which uses timing and phase analysis for detecting radio types. Second, we compare against Jello [28] which uses edge detection on the FFT to detect occupied spectrum. Finally, we compare against SecureAngle's [27] MUSIC technique [23] for computing AoA.

Testbed: The testbed for the experimental results consists of an indoor office environment with cubicle-style office rooms (see Fig.5). The total office size was 105ft $\times$ 48ft, the ceiling height was 10ft, and

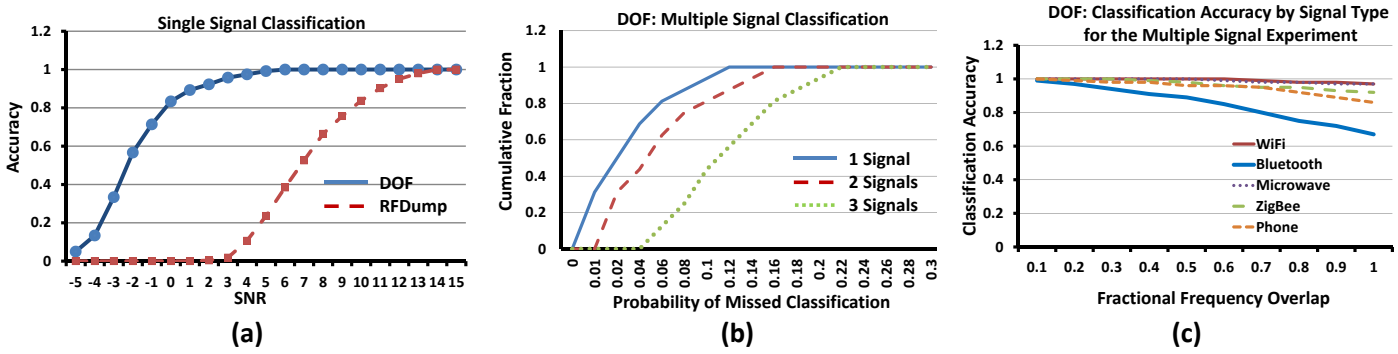

Figure 6: DOF has high classification accuracy over a large range of SNRs and for multiple interfered signals.

the height of the cubicle partitions was 5.5ft. Our wideband radio was placed at three different locations as shown by the shaded circles in Fig. 5, while the radios that we wish to detect (WiFi, Zigbee, cordless phones, bluetooth devices and microwaves) are placed randomly in the office and allowed to transmit. The measurements were taken when the office was empty, and ambient interference from sources outside our control (the departmental WiFi network, microwaves etc) was absent. While the design of DOF was tested using data from a 100 MHz channel sounder, note that conceptually DOF will work with any stream of raw data which can be obtained via commodity software radios such as USRPs. We used the channel sounder as opposed to USRPs because we wanted to demonstrate the full breadth of DOF's capabilities and did not want the range of our tests to be curbed by the limitations of the data acquisition device.

Training Data: The DOF SVM classifiers are first trained with labeled data generated via controlled experiments in the testbed. The training signals are generated by randomly turning on one of the five radios at a random location with randomly picked PHY parameters when applicable (bitrate, channel etc). Turning on means continuously transmitting packets for WiFi, Zigbee and Bluetooth radios, making a continuous call for the cordless phone and powering on for the microwave oven. We generate 30 labeled points for each radio type. The SVMs for detecting multiple signal types are trained by synthetically combining the single signal labeled data as described in Sec. 4.2. Hence the training complexity of DOF is relatively modest. Note that once DOF is trained, the training data allows DOF to operate in any physical environment so long as the training is representative of all possible parameters that a signal could have (bitrate, modulations, etc.). But for signals which aren't FCC-certified (e.g. microwaves), training has to be done specific to each instance since those protocols are not governed by a uniform specification.

Calculating SNR: In our plots, the reader will often see measurements at SNRs as low as -5dB. The reason we are able to calculate such low SNRs is our wideband radio, which is a modified channel sounder. Specifically, the sounder was initially designed to conduct wide area surveying for a WiMax network deployment. In such scenarios, such low SNRs need to be measured and the sounder comes equipped with a proprietary technique that allows two sounders to be placed at separate locations and yet accurately measure the SNR between them even when it is as low as -5dB. We leverage this capability to measure the SNRs in our experiments.

6.1 Estimating Signal Types

We evaluate DOF's accuracy in detecting component signal types in the received signal and compare it to the accuracy of RFDump [16], defining accuracy as the probability of correct classification.

Method: For each run, we pick a random subset of the five different radio types. We place the corresponding radios at a random location, randomly set their PHY parameters (bitrate, channel etc) in the testbed and allow them to transmit. We also measure the SNR of the channel from each location. The same received samples are passed to the DOF detector and the RFDump detector - both algorithms are run at the same bandwidth. Because RFDump was not designed for such a large bandwidth, it does not work if there are multiple signals overlapping in time and for legacy radios such as cordless phones and microwave ovens. Hence for RFDump, we eliminate traces with multiple interfered signals, or if they have analog phone or microwave signals in them and compute its accuracy only for the remaining three signal types. Fig. 6 plots the accuracy of DOF and RFDump against SNR when there is a single signal. Because SNR doesn't work as a metric when there are multiple interfered signals, we plot the CDF of the error across all experimental runs in Fig. 6.

Analysis: Figs. 6(a) and 6(b) show that DOF has high classification accuracy over a large range of SNRs and for multiple interfered signals. DOF achieves an accuracy ranging from $85-100\%$, even for SNRs as low as 0dB when there is a single signal present. For multiple interfered signals, DOF achieves an accuracy greater than 85% at least 90% of the time even when there are three interfered signals in the trace. DOF is robust to SNR because our feature vector components are calculated by correlating and integrating repeated patterns over long intervals, hence even if individual samples have low power, the integration over the entire interval yields very prominent features. Also since the repeating patterns are unique to each signal and uncorrelated with other signal types, they are quite robust to the presence of interfering signals.

RFDump achieves an accuracy of at most 60% when the SNR of the detected signal is between between -5 to 8dB. RFDump uses two techniques, timing analysis and phase analysis to classify signal types. Timing analysis is based on detecting start and end of packets using energy detection, while the phase analysis component is dependent on computing statistics of the phases of received samples which use phase modulation such as Zigbee and Bluetooth. Both operations are error prone at medium to low SNRs, since noise significantly affects the accuracy of energy detection, and distorts the received phases affecting the phase statistics. Finally, RFDump fails to work in the presence of multiple interfering signals, since it cannot detect start or end of packets reliably when signals overlap in time, and phases are distorted when there is a strong additive interferer.

Why is accuracy slightly lower for interfering signals? DOF's accuracy is slightly lower when there are multiple interfering signals present in the received signal. Because DOF's ability to classify multiple interfering signals hinges on how well it is able to exploit asynchrony, at first glance it seems like this may be the root of the problem. Upon closer inspection, we found that while asynchrony detection errors are present, they account for a small fraction of the overall errors. Asynchrony detection errors occur when the offset between different transmissions is shorter than the cyclic feature pro-

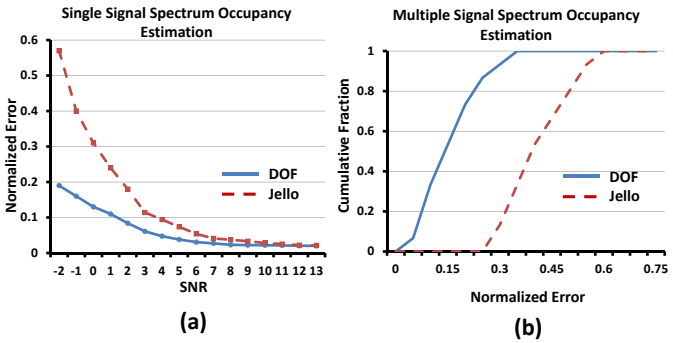

Figure 7: DOF is more accurate than edge detection at low SNRs and with multiple interfering signals in estimating occupied spectrum

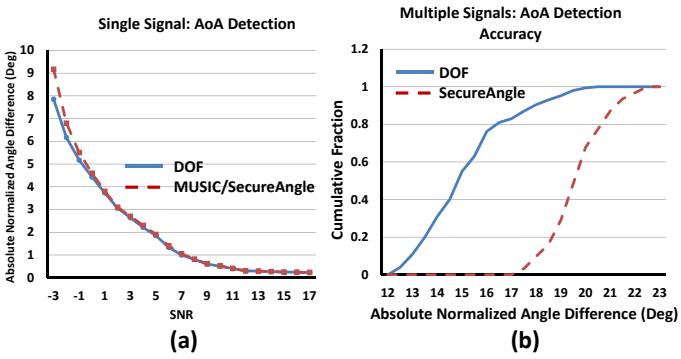

Figure 8: AoA Estimation Accuracy - Single Signal accuracy is accurate even at low SNRs but when multiple radios are operating, there are often more significant AoA's than our detectors are able to discern, a fundamental limit due to the number of antennas in our system.

cessing time. Because an FFT of length 512 using FFTW [1] can be performed in $4\mu s$ and the extra complexity of cyclic feature extraction is $< 30\%$ over that of a standard FFT (sec. 5), the probability of repeatedly missing the offset between asynchronous transmissions (WiFi, ZigBee, and Bluetooth all have packet lengths on the order of 100's of μs to a few ms) is small.

The main reason for the lower accuracy is that certain low-power and low-bandwidth signals are not detected in some corner cases when there is strong frequency overlap. Fig. 6 plots the accuracy of our classification for different signal types as a function of the frequency overlap from another signal. Frequency overlap between signals of the same type is rare because their identical MAC protocols act as a mechanism to prevent this. When signals do overlap, this is because they are of different types. Referring to Table 1, different types of signals centered at the same carrier frequency exhibit distinct patterns. Thus overall, DOF's classification accuracy does not deteriorate drastically when signals overlap, except for Bluetooth, which we can see drops with increasing overlap. The reason is that Bluetooth signals have a low bandwidth of 1 MHz. If the signal is overlapped in frequency by a stronger signal like WiFi, then DOF fails to even detect the Bluetooth signal. Bluetooth signals only have a few unique features because of their simple structure and small bandwidth, while a WiFi signal has a rich feature set some of which are close to the Bluetooth pattern frequencies. Consequently, DOF ends up not detecting the Bluetooth signal, resulting in lower accuracy.

6.2 Estimating Occupied Spectrum

In this section, we evaluate the accuracy of DOF's spectrum occupancy estimation, and compare it with the edge detection based approach in Jello [28]. To make a fair comparison, we allow Jello to use the same 512 length FFT as DOF.

Method: The experiment is conducted similar to the above classification experiments and the raw dump of the received signal at our wideband radio is sent to DOF's and Jello's spectrum occupancy estimators. We take the estimated occupied spectrum from both systems, and compute the absolute error for both. The error is computed as the sum of the estimated occupied spectrum components that are not actually occupied plus the estimated unoccupied spectrum which is actually occupied. We normalize the error by the ground truth spectrum occupancy. We plot two separate figures, Fig. 7(a) plots the error vs SNR of the detected signal when there is a single received signal in the trace and Fig. 7(b) plots the CDF of normalized errors when there are more than 1 potentially overlapping signals in the trace.

Analysis: Fig. 7(a) shows that DOF is reasonably accurate in estimating occupied spectrum. The normalized error in estimating occupied spectrum is around 15% at low SNR and reduces to 5% at higher SNR, but never approaches 0 because of the FFT size which inherently limits resolution.

Fig: 7(b) plots the CDF of errors when there is more than one signal in the received trace. DOF achieves a median error of 15% in these experiments, slightly higher than the single sender case. Apart from the FFT resolution, the other contributor to the error is overlapping signals in the frequency domain. As in the classification case, when a strong signal overlaps in frequency with a weak signal like Bluetooth, it becomes hard to even detect that the Bluetooth signal exists and consequently we miss its feature vector components. Hence, the spectrum occupancy error is slightly higher.

Jello performs less accurately, especially at low SNR and with multiple interfering signals. The reason is that edge detection (the technique used in Jello) is based on computing the slope of the PSD. However, at low SNRs noise introduces sufficient fluctuations that we encounter large slopes in the derivative of the signal at frequencies away from where the transmitted signal lies. Further, edge detection can get confused when there are two partially overlapping signals in frequency. The reason is that when the overlap ends, there will be a sharp drop in the PSD level (because we went from two signals to one signal at that frequency). This can be mistaken to be the end of the occupied spectrum since the PSD is relatively flat after that transition. Consequently, as we see in Fig. 7 Jello has a higher median error of 40% in our experiments.

6.3 Estimating Angle of Arrival

In this section, we evaluate the accuracy of DOF's AoA estimation component. However, unlike the prior experiments, we cannot compare against ground truth here. The indoor environment is a multipath environment, and a transmitted signal can arrive at multiple angles simultaneously. We have no way of knowing exactly what scattering takes place and consequently the ground truth AoAs. Hence we conduct the experiment as follows: We use two of our wideband radios, one equipped with 4 antennas and another with 8 antennas arranged in a ULA. As in the previous experiments, we randomly pick a subset of the radios among our five different types, place them at a random location and let them transmit. For the trace from the 8-antenna radio, we apply the standard MUSIC technique [23] to estimate all AoAs. The reason is that with such a large antenna array, MUSIC is almost guaranteed to accurately find all the significant AoAs. We consider these angles to be the ground truth.

Next, we give the trace collected at the 4-antenna radio to DOF as well as SecureAngle's [27] MUSIC method. Our logic for picking 4 antennas is to make it consistent with state of the art MIMO hardware, which comes with around 4 antennas. We then compute the absolute error of the estimates from DOF and SecureAngle, which is computed by summing the following values: absolute value of each estimated angle minus the closest ground truth angle. The absolute error is

normalized by the number of estimated angles. Fig. 8(a) plots the normalized angle error vs the SNR when the trace contains a single signal type, while Fig. 8(b) plots the CDF of normalized errors when it contains more than one signal.

Analysis: Fig. 8(a) shows that DOF computes the AoAs with an accuracy of at least 5 degrees even at low SNRs when there is a single signal. SecureAngle's accuracy is similar. The reason for the relatively worse performance at very low SNR is that the estimation algorithm uses projections of the Spectral Correlation Function matrix to compute angles of arrival, and the projections have a slight contribution from noise. At very low SNRs, the contribution is relatively significant, and hence causes a higher estimation error.

As we see in Fig. 8(b) both DOF and SecureAngle perform slightly worse when there are multiple signals. DOF's median error is around 14 degrees, while SecureAngle's is 19 degrees. The reason is that the number of AoAs that can be accurately detected is a function of the number of antennas a radio has. With 4 antennas, we can detect at most 4 significant angles of arrival [23]. However with multiple signals in a rich multipath environment, there will be significant signal strength along a number of angles, sometimes larger than 4. Both DOF and SecureAngle get confused in this case. However, we note that this is a fundamental problem [23], regardless of the algorithm, the number of antennas a node has places a sharp upper bound on how many AoAs can be distinguished.

7. APPLICATION TO SMART RADIOS

The most direct uses of DOF are in designing smart radios, network management, indoor localization and performance diagnosis. While we leave most of these to future work, we design DOF-SR, a preliminary prototype of a wideband smart radio to demonstrate the benefits of DOF. Our design is inspired by recent work in smart radios, including Jello [28] and others [22]. We compare DOF-SR with Jello [28], the most recent state of the art system for such designs.

7.1 DOF-SR

DOF-SR is a wideband policy-aware smart radio design that operates over the entire 100MHz ISM band. The key technical contribution in DOF-SR is its ability to take advantage of the accurate detecting substrate DOF provides to let users specify a policy that tunes how aggressive the radio is going to be in scavenging for spectral resources. To demonstrate the policy flexibility, we design three sample policies and implement them in our current prototype of DOF-SR

1. **P0**: Only use unoccupied spectrum.
2. **P1**: Use all unoccupied spectrum. Further use spectrum occupied by microwave oven radiation.
3. **P2**: Use all unoccupied spectrum as well as parts occupied by microwave oven radiation. Further, compete for spectrum occupied by WiFi radios and get half the time share on that part of the spectrum.

The three policies are ordered in increasing amounts of aggressiveness. The first policy plays it safe and is similar to the one used by Jello. The second is more aggressive, but still avoids harming any co-existing radio that is used for communication. The third is the most aggressive, and encodes the notion that since WiFi is also another unlicensed radio, it is fair to compete and obtain half the time on spectrum used by WiFi too. However, our key point is that there is no "universal right policy", it will depend on the user's preferences and environmental constraints, but DOF-SR provides the flexibility needed to adapt the policy to those preferences and constraints.

7.1.1 Protocol

Measuring the RF Neighborhood: DOF-SR uses DOF as the substrate to accurately measure the RF environment and create a *RF-profile*. In our design, both DOF-SR sender and receiver radios measure the environment using DOF, and the receiver sends its measurements to the sender. The measurement consists of the 2-D profile of the RF environment along the frequency and spatial (AoA) axes that DOF estimates, with each occupied point annotated by the occupying signal type. The sender combines the measurements from the receiver by taking the union of both spectrum occupancy measurements, but uses the AoA estimates from the receiver since AoA is specific to the detecting radio and only matters at the receiver for communication.

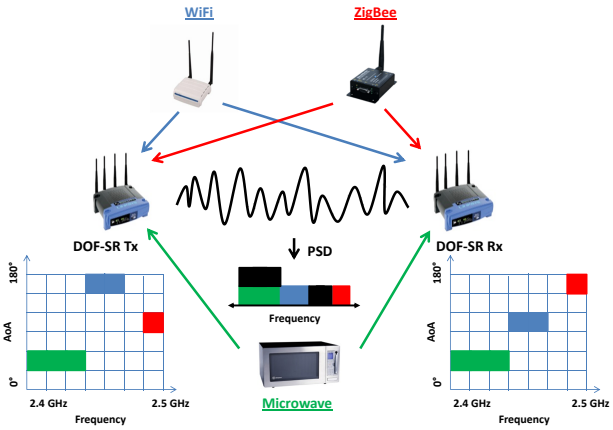

Figure 9: Evaluated scenario for DOF-SR

Estimating what spectral resources to use: Next, the sender uses the merged RF profile along with the user specified policy to estimate what spectral resources to use. For policy **P0**, this would be only the unoccupied spectrum, while for **P1** and **P2** this would also include spectrum occupied by microwave ovens and WiFi respectively.

Creating Packets: DOF uses an OFDM-MIMO PHY layer to create its packets for transmission. The key challenge here is to adaptively leverage the 4 antenna MIMO frontend to maximize throughput while minimizing interference from and to the co-existing radios. We first discuss how the system would work for the simplest policy **P0**, and then extend it to work for the other two policies.

Our current OFDM implementation uses a 1024 point FFT, and divides the 100MHz band into 1024 subcarriers of length 96KHz each. Among these subcarriers, it marks all subcarriers that intersect with the occupied parts from the RF profile as unusable. On the remaining subcarriers it uses MIMO spatial multiplexing to transmit 4 independent streams on each subcarrier. We omit the details here, but refer the reader to [26] for a description of this standard technique.

For policies **P1** and **P2**, we modify the above algorithm to take advantage of their aggressiveness. Specifically, for **P1** we include the subcarriers that were detected to be occupied by microwave ovens in the RF profile. However, we cannot use spatial multiplexing on these subcarriers, since the interference from the microwave signals would be too strong. Instead we leverage the 4 MIMO antennas to perform beamforming and null the interference from the microwave oven signals. Specifically, lets say the microwave oven signals are arriving at i significant AoAs $\theta_1, \ldots, \theta_i$ at the DOF-SR receiver. The sender calculates antenna weights $\vec{w}_p^S = (w_1^S, w_2^S, w_3^S, w_4^S)$ for the subcarrier centered at f_p, which is interfered by the microwave oven, such that the transmitted signal will not arrive at the same angles as the microwave. The receiver will then calculate antenna weights antenna weights $\vec{w}_p^R = (w_1^R, w_2^R, w_3^R, w_4^R)$ such that the microwave signal from the estimated AoAs at the receiver will be minimized:

$$\arg_{\vec{w_p}} \min |\vec{w}_p^* \phi(\vec{\theta}_1) + \ldots + \vec{w}_p^* \phi(\vec{\theta}_i)| \quad (11)$$

where $\phi(\vec{\theta}_i)$ is the AoA vector corresponding to θ_i at the 4 antennas defined in Eq. 10, and $\vec{w}_p^*$ is the conjugate transpose. The estimated

antenna weights are then applied to the streams on the corresponding OFDM subcarriers.

For policy **P3**, the DOF-SR radio will time share the medium with the co-existing WiFi radio, i.e. it will transmit on that spectrum half the time. The key parameter here is the time period over which the smart radio transmits and stays idle. If the time period is too short, then the WiFi radio won't have enough time to accurately estimate the bitrate and correctly utilize its channel time. In our current implementation we use a conservative period of 200ms, since that gives an 802.11 WiFi radio enough time to estimate the channel and get nearly 80 packets through even at the lowest bitrate. Hence, DOF-SR uses the WiFi spectrum for 200ms and then stays away for 200ms. During the time it uses that spectrum, DOF-SR uses spatial multiplexing on all 4 antennas.

Packet Transmission: Before transmitting the encoded packet, a DOF-SR sender transmits a short control packet over a predefined narrowband control channel to the receiver to synchronize state. This packet contains information on what subcarriers will be used, and the antenna weights on the used subcarriers. Then the sender transmits the packet and waits for an ACK, and repeats the above process.

Caveats: The goal of our current DOF-SR implementation is to show the potential benefits of a smart PHY that leverages the detection capabilities of DOF. Hence, it does not tackle MAC layer issues such as finding a usable control channel, rate adaptation on the used spectrum and contention among multiple DOF-SR nodes. The full design and implementation of a smart radio network stack based on DOF is beyond the scope of this paper and is part of our future work. However, the current prototype suffices to evaluate the relative benefits of DOF-SR over the compared state of the art approaches.

Compared Approach: We compare with Jello [28], which is a smart radio design that estimates unused spectrum using edge detection and allocates them among multiple radios for communication. We implement Jello also on our wideband radio. To make a fair comparison, since DOF-SR weaves non-contiguous spectrum together, we modify Jello to also weave non-contiguous spectrum using OFDM. Further, since we are using spatial multiplexing with 4 antennas, we let Jello also use the same spatial multiplexing capabilities with 4 MIMO antennas. Thus the only differences between DOF-SR and Jello in our current implementations are that DOF-SR uses DOF as its detector, while Jello uses edge detection. Second, DOF-SR with policies **P1** and **P2** uses microwave oven and WiFi occupied spectrum appropriately, while Jello does not since edge detection cannot detect that it is a microwave oven or WiFi occupied spectrum. Clearly, both these extra capabilities for DOF-SR come because of the DOF detector, and therefore help us quantify the benefits of using DOF.

Metric: We cannot use throughput as the metric to compare the two designs, since a naive scheme that uses no detection will always achieve the maximum throughput, but harmfully interfere with all co-existing radios. The right metric is therefore one that allows us to visualize the tradeoff between throughput and the harmful interference which the smart radio causes to co-existing radios. To evaluate this tradeoff, we compute two quantities and plot them against each other:

- **Normalized Throughput:** We compute the throughput achieved by DOF-SR and Jello, and normalize them by the throughput an optimal offline scheme implementing our policy would achieve. To compute the throughput of the optimal scheme, we take advantage of the fact that we know the ground truth of what radios are operating and what spectrum they are occupying. We also feed it the AoA measurements from DOF, since we cannot know the ground truth due to unknown multipath effects. We then use this information to compute the throughput of the optimal scheme which would use exactly the unoccupied spectrum, and optimally beamforms its signals in the microwave occupied spectrum.
- **Normalized Harmful Interference:** We measure the throughput of the co-existing WiFi and Zigbee radios (which are supposed to be protected according to our policy) when neither DOF-SR or Jello are operating, and then when they are operating. We compute the difference in throughput, and normalize it by the throughput they achieve when the smart radios are not operating. This quantity represents the normalized performance drop due to the operation of DOF-SR or Jello.

The ideal scheme would have a normalized throughput of 1 and a normalized harmful interference of 0.

7.2 Evaluation

We evaluate DOF-SR and Jello on the same indoor testbed described in Sec. 6.

Method: We randomly place a WiFi sender-receiver pair, a Zigbee sender-receiver pair and a microwave oven in the testbed. The WiFi and Zigbee radios are operating on randomly picked non-intersecting channels, and the bitrate depends on their respective channel conditions. The WiFi and Zigbee links are continuously transmitting packets. We take a raw 10 second dump using our wideband radios at the sender and receiver, and provide the dumps to DOF and the edge detection algorithm of Jello. After their respective computations, we let the two smart radio systems compute what spectral resources they are going to use and how. They are then allowed to transmit one after the other for 50 seconds. For DOF-SR, we transmit three separate times corresponding to the three policies. We then take the traces at the receiver, decode the signals and compute the goodputs. Simultaneously, we measure the throughput of the WiFi and Zigbee links. We compute the normalized throughput and the normalized harmful interference as discussed before. We repeat this experiment for 50 such configurations and plot the points in Fig. 10.

Analysis: Fig. 10 plots normalized throughput on the y-axis and normalized harmful interference to the co-existing WiFi and Zigbee links on the x-axis for DOF-SR and Jello. We have three plots, corresponding to the three policies that DOF-SR currently implements. The blue circles are for DOF-SR, and the red crosses are for Jello. Note that the optimal scheme will achieve a normalized throughput of 1 and normalized harmful interference of 0.

We first summarize the results:

- With policy **P0**, DOF-SR achieves an average normalized throughput of 0.93 and the average harmful interference it causes is around 0.1, i.e. the throughput of the WiFi and Zigbee links drop only by around 10%. Jello on the other hand achieves a normalized throughput of 0.82 and causes a harmful interference of 0.44. Thus DOF-SR gets a gain of 15% over Jello purely from more accurate unoccupied spectrum estimation, and causes 35% less harm than Jello.
- With policy **P1**, DOF-SR achieves an average normalized throughput of 0.93 and causes an average harm of 0.1. Jello of course cannot use this policy and consequently its average normalized throughput drops to 0.61, while its average harm stays the same at 0.44. Thus with the ability to use microwave oven occupied spectrum, DOF-SR provides a 50% increase over Jello, while still causing minimal harm to co-existing radios.
- With policy **P2**, DOF-SR achieves an average normalized throughput of 0.87 and causes an average harm of 0.32. The reason for the higher harm is that DOF-SR is now competing with the WiFi device for half the time on the WiFi occupied spectrum. Hence WiFi throughput naturally drops compared to **P1**, and harm increases. However this is intended, policy **P2** was designed to be aggressive and steal throughput from the WiFi

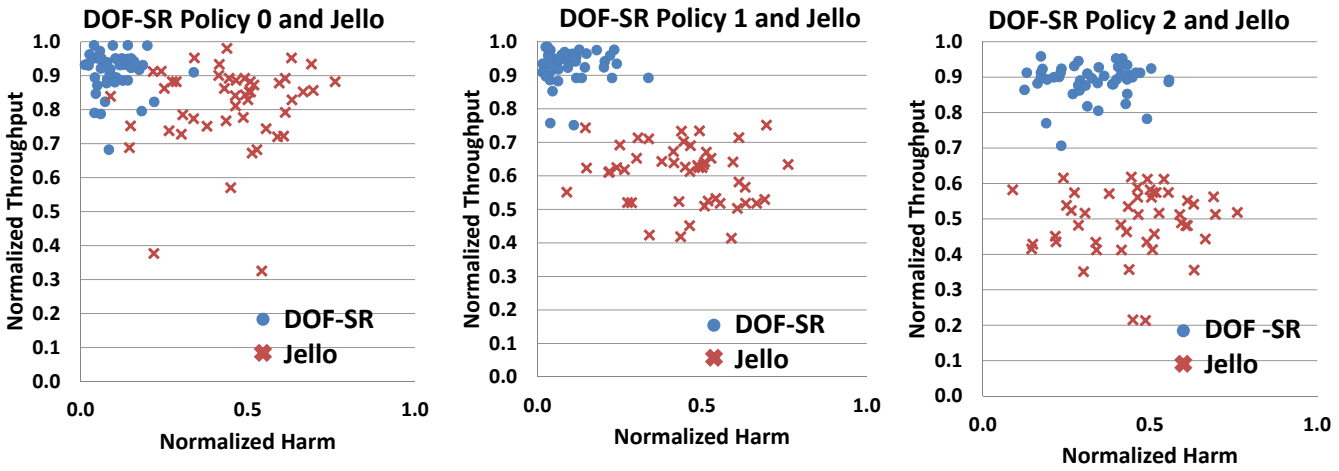

Figure 10: Throughputs and Harm with Smart Radios - More aggressive policies enable higher throughput but also cause greater harm to legacy systems. WiSpy-SR enables users to decide how aggressive their policy should be.

node. The normalized Jello throughput is around 0.5 with the harm the same at 0.44. Hence with policy **P2**, DOF-SR provides a performance gain of nearly 80% over Jello.

The relative gains over Jello help us understand the gains the increasingly aggressive policies provide to DOF-SR. The gain of 15% with policy **P0** is purely from DOF's more accurate spectrum detection. Further note that we achieve this gain while causing minimal harm to the co-existing radios, their average throughput loss is less than 10%. Next, with policy **P1**, DOF-SR's gains increase by another 35% to 50%. This gain comes from DOF-SR's extra capability of being able to detect microwave oven signals and their AoAs, and leverage that information to beamform and null the interference to increase throughput. Finally, with policy **P2**, DOF-SR gets a gain of 80% over Jello, i.e. an additional 30% over policy **P1**. However, the gains come with the price of increased interference to the co-existing WiFi radio since when competing DOF-SR is likely to cut the WiFi throughput, and hence the harm increases to 32%.

8. CONCLUSION

Historically, unlicensed band co-existence has been managed "socially". Different protocols would largely use non-overlapping bands, and given the low density of radios in a neighborhood, the likelihood of radios stepping on each other's toes was low. However, with the increasing number of protocols that operate in the ISM band and the increasing density of radios around us, this assumption is more and more on shaky ground. DOF provides the accurate substrate that future ISM band radios would need to operate and co-exist in this crowded space. DOF opens up a number of avenues of future work, including designing a generalized policy-aware smart radio, whose preliminary prototype design we briefly described in this paper. We also plan to apply DOF to other applications in network management, performance diagnosis and indoor localization.

9. REFERENCES

[1] Fftw. `http://www.fftw.org`.

[2] Fftw benchmark. `http://www.fftw.org/benchfft`.

[3] Gnu radio. `http://gnuradio.org`.

[4] Practical signal detection and classification in gnuradio. 2007.

[5] C. Bishop. Pattern recognition and machine learning, 2006.

[6] J. Capon. High-resolution frequency-wavenumber spectrum analysis. *Proceedings of the IEEE*, 57(8):1408 – 1418, aug. 1969.

[7] M. Davy, A. Gretton, A. Doucet, and P. Rayner. Optimized support vector machines for nonstationary signal classification. *Signal Processing Letters, IEEE*, 9(12):442 – 445, dec. 2002.

[8] A. Fehske, J. Gaeddert, and J. Reed. A new approach to signal classification using spectral correlation and neural networks. pages 144 –150, nov. 2005.

[9] W. Gardner. Exploitation of spectral redundancy in cyclostationary signals. *Signal Processing Magazine, IEEE*, 8(2):14 –36, apr. 1991.

[10] S. Gollakota and D. Katabi. ZigZag decoding: combating hidden terminals in wireless networks. In *SIGCOMM '08: Proceedings of the ACM SIGCOMM 2008 conference on Data communication*, pages 159–170, New York, NY, USA, 2008. ACM.

[11] S. Haykin, D. Thomson, and J. Reed. Spectrum sensing for cognitive radio. *Proceedings of the IEEE*, 97(5):849 –877, may. 2009.

[12] A. P. Iyer, K. Chintalapudi, V. Navda, R. Ramjee, V. N. Padmanabhan, and C. R. Murthy. Specnet: spectrum sensing sans frontières. In *Proceedings of the 8th USENIX conference on Networked systems design and implementation*, NSDI'11, pages 26–26, Berkeley, CA, USA, 2011. USENIX Association.

[13] L. Jia, Y. Gao, J. Isoaho, and H. Tenhunen. A new vlsi-oriented fft algorithm and implementation. In *IEEE International ASIC Conference*, 1998.

[14] D. H. Johnson and D. E. Dudgeon. *Array Signal Processing: Concepts and Techniques*. Simon & Schuster, 1992.

[15] M. O. C. R. L. Bixio, G. Oliveri. Ofdm recognition based on cyclostationary analysis in an open spectrum scenario. In *Vehicular Technology Conference*, 2009.

[16] K. Lakshminarayanan, S. Sapra, S. Seshan, and P. Steenkiste. Rfdump: an architecture for monitoring the wireless ether. In *Proceedings of the 5th international conference on Emerging networking experiments and technologies*, CoNEXT '09, 2009.

[17] E. Like, V. D. Chakravarthy, P. Ratazzi, and Z. Wu. Signal classification in fading channels using cyclic spectral analysis. *EURASIP J. Wirel. Commun. Netw.*, 2009:3–3, 2009.

[18] G. V. L. J. N. Czink, B. Bandemer and A. Paulraj. Stanford july 2008 radio channel measurement campaign. In *COST 2100*, October 2008.

[19] K. E. N. P D Sutton and L. E. Doyle. Cyclostationary signatures for rendezvous in ofdm-based dynamic spectrum access networks. In *IEEE DySPAN*, 2007.

[20] T. M. R. M. Paramvir Bahl, Ranveer Chandra and M. Welsh. White space networking with wi-fi like connectivity. In *ACM SIGCOMM*, 2009.

[21] K. N. Paul Sutton and L. Doyle. Cyclostationary signatures in practical cognitive radio applications. *IEEE Journal on Selected Areas of Communications*, 26, Jan 2008.

[22] H. Rahul, N. Kushman, D. Katabi, F. Edalat, and C. Sodini. Narrowband friendly wideband radios. In *ACM SIGCOMM 2008*.

[23] R. Schmidt. Multiple emitter location and signal parameter estimation. *Antennas and Propagation, IEEE Transactions on*, 34(3):276 – 280, mar. 1986.

[24] A. Swami and B. Sadler. Hierarchical digital modulation classification using cumulants. *Communications, IEEE Transactions on*, 48(3):416 –429, mar. 2000.

[25] T. Taher, M. Misurac, J. LoCicero, and D. R. Ucci. Microwave oven signal modeling. 2008.

[26] D. Tse and P. Vishwanath. *Fundamentals of Wireless Communications*. Cambridge University Press, 2005.

[27] J. Xiong and K. Jamieson. Secureangle: Improving wireless security using angle-of-arrival signatures. In *ACM HotNets 2010*.

[28] L. Yang, W. Hou, L. Cao, B. Y. Zhao, and H. Zheng. Supporting demanding wireless applications with frequency-agile radios. In *USENIX NSDI*, April 2010.

[29] O. Zakaria. Blind signal detection and identification over the 2.4ghz ism band for cognitive radio. In *Master's thesis, University of South Florida, Tampa, Florida, USA*, 2009.